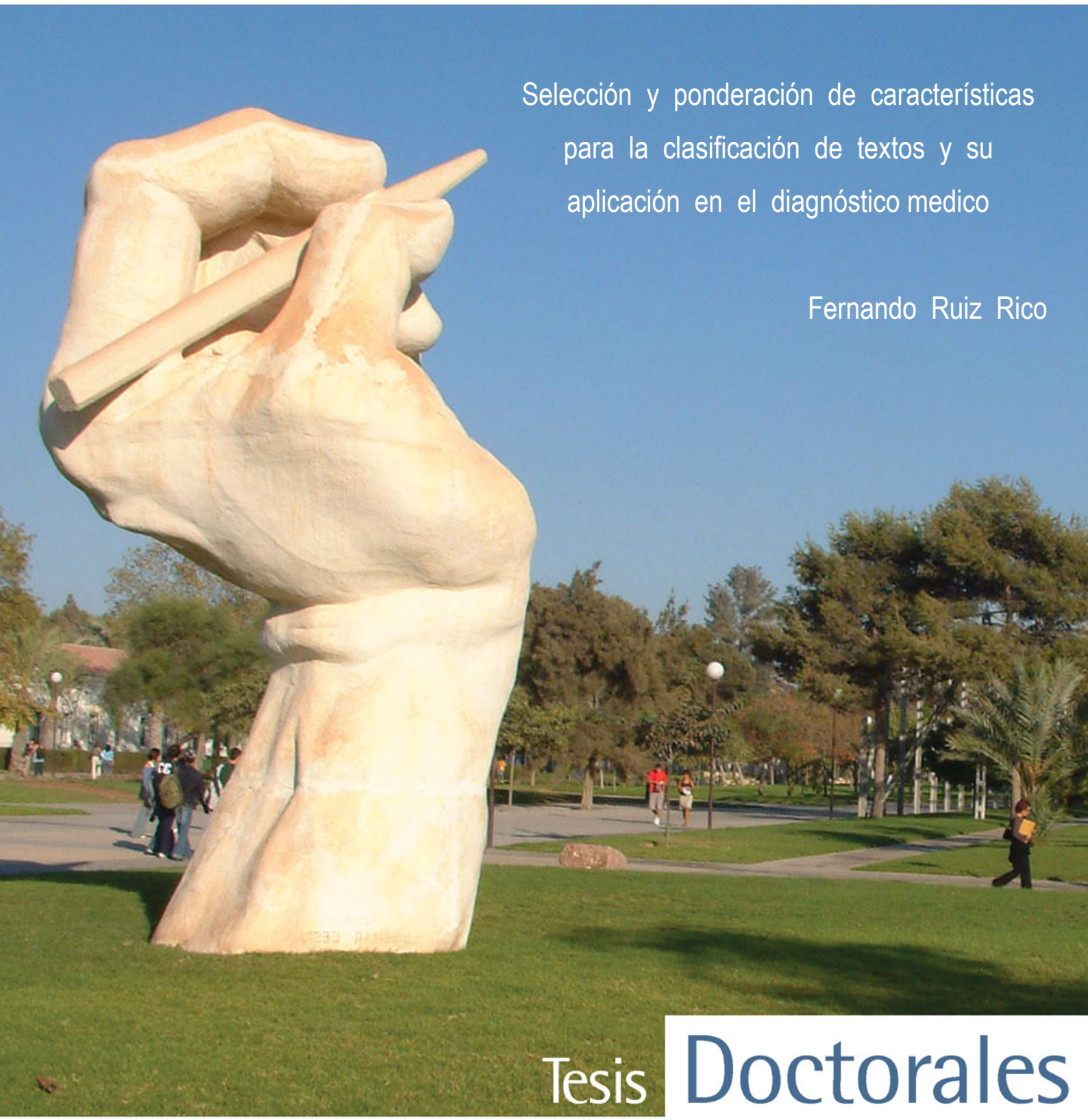
Universitat d'Alacant
Universidad de Alicante

Selección y ponderación de características para la clasificación de textos y su aplicación en el diagnóstico medico

Fernando Ruiz Rico



Tesis Doctorales

www.eltallerdigital.com

UNIVERSIDAD de ALICANTE

TESIS DOCTORAL

Selección y ponderación de características para la clasificación de textos y su aplicación en el diagnóstico médico

Fernando Ruiz Rico

frr@alu.ua.es

Directores

Dr. José Luis Vicedo González

Dr. David Tomás Díaz

Universitat d'Alacant
Universidad de Alicante

Universitat d'Alacant
Universidad de Alicante

Julio 2013

Universitat d'Alacant
Universidad de Alicante

Agradecimientos

Gracias a Pedro Pastor, por el primer empujón.

Gracias a José Luis Vicedo, por tanta paciencia que ha demostrado tener conmigo durante todos estos años.

Gracias a David Tomás, por aparecer siempre en el momento adecuado con esas palabras que faltaban.

Gracias a mis padres, porque la ilusión no se aprende, se transmite.

Gracias a Jose, Dani y Sandra, por recordarme que siempre hay motivos para seguir adelante.

Gracias a Marcos y a David, por sus sonrisas, sus besos y sus abrazos, y por enseñarme a verlo todo desde otro punto de vista.

Y gracias especialmente a Mari Chelo, por todo ese tiempo nuestro que me he quedado yo.

Julio 2013 Fernando Ruiz Rico

Universitat d'Alacant
Universidad de Alicante

Universitat d'Alacant
Universidad de Alicante

Índice general

Índice general I

Índice de figuras VII

Índice de tablas XI

1. Introducción 1

1.1. Motivaciones 4

1.1.1. El problema de la dimensionalidad 5

1.1.2. Relevancia de los términos 6

1.1.3. La clasificación de textos en el mundo real 6

1.2. Objetivos 8

1.3. Metodología 9

1.4. Estructura de la tesis 10

2. La clasificación de textos 13

2.1. Tipos de clasificación 14

2.1.1. Clasificación binaria vs. clasificación multiclase 14

2.1.2. Pivote en el documento vs. pivote en la categoría 14

2.1.3. Asignación directa vs. ranking de categorías 15

2.2. Aproximaciones al diseño de clasificadores 16

2.2.1. Sistemas basados en conocimiento 17

2.2.2. Sistemas basados en aprendizaje automático 18

Universitat d'Alacant
Universidad de Alicante

2.3. Componentes de un clasificador 20
2.3.1. Taxonomías 21
2.3.2. Corpus de trabajo 22
2.3.3. Características de aprendizaje 25
2.3.4. Algoritmos de clasificación 35
2.4. Aplicaciones de la categorización de textos 40
2.4.1. Filtrado de textos 42
2.4.2. Clasificación de noticias 42
2.4.3. Categorización jerárquica de páginas web 43
2.4.4. Identificación del idioma 43
2.4.5. Desambiguación del sentido de las palabras 44
2.4.6. Etiquetado de entidades 44
2.4.7. Clasificación de preguntas para sistemas de búsqueda de respuestas . 44

3. Trabajos previos **47**

3.1. Representación de documentos 48
3.1.1. N-gramas de letras 49
3.1.2. Palabras 49
3.1.3. N-gramas de palabras 50
3.1.4. Otras características 50
3.2. Reducción de la dimensionalidad 51
3.2.1. Métodos de selección de características 51
3.2.2. Métodos de extracción de características 55
3.2.3. Selección estadística 58
3.2.4. Combinación de métodos 58
3.3. Ponderación de términos 59
3.3.1. Factor local 60
3.3.2. Factor global 62
3.3.3. Factor de normalización 69

Universitat d'Alacant
Universidad de Alicante

3.4. Algoritmos de clasificación 70
3.4.1. Vecinos más cercanos (kNN) 70
3.4.2. Rocchio 72
3.4.3. Máquinas de vectores de soporte (SVM) 73
3.4.4. Naive bayes (NB) 75
3.4.5. Clasificadores unificados 77
3.4.6. Aproximaciones basadas en recuperación de información 78
3.5. Corpus y taxonomías 79
3.5.1. OHSUMED 79
3.5.2. Reuters 80
3.6. Escalabilidad 80
3.6.1. Palabras vs. frases 81
3.6.2. Dimensionalidad y cálculo de pesos 82
3.6.3. Vectores de soporte, vecinos y prototipos 83

4. Ponderación y selección de características **87**
4.1. Características de aprendizaje 88
4.1.1. Descriptores de categorías 88
4.1.2. Sustantivos y adjetivos 88
4.1.3. Selección de términos válidos 90
4.2. Ponderación de características 91
4.2.1. Localización dentro del documento 92
4.2.2. Particularización por categoría 92
4.2.3. Normalización de pesos 93
4.2.4. Cálculo de TF, DF y CF 93
4.2.5. Cálculo de la relevancia 95
4.3. Reducción de la dimensionalidad 98
4.3.1. Discriminación basada en la media 98
4.3.2. Estrategias de reducción 98

Universitat d'Alacant
Universidad de Alicante

4.4. Algoritmo de clasificación 100
4.4.1. Generación de vectores representativos de categorías 101
4.4.2. Clasificación mediante suma de pesos 103
4.4.3. Algoritmos no basados en prototipos 103

5. Evaluación **105**
5.1. Corpus de trabajo 105
5.1.1. Reuters-21578 107
5.1.2. OHSUMED 108
5.2. Medidas de evaluación 110
5.2.1. Precisión y cobertura 111
5.2.2. Punto de equilibrio entre precisión y cobertura (BEP) 113
5.2.3. F1 113
5.2.4. Precisión media (AvgP) 114
5.2.5. One-Error (OneErr) e Is-Error (IsErr) 115
5.2.6. Micropromedio y macropromedio 116
5.3. Experimentos 117
5.3.1. Selección de características 118
5.3.2. Ranking de categorías 123
5.4. Relevancia de los parámetros en el proceso de clasificación 129
5.4.1. Ponderación y selección de características 130
5.4.2. Ranking de categorías 131
5.4.3. Peso óptimo para las expresiones de los títulos 133

6. Aplicaciones **139**
6.1. Diagnóstico médico 139
6.1.1. Corpus de trabajo 141
6.1.2. Medidas de evaluación 142
6.1.3. Funcionamiento rápido y funcionamiento por jerarquías 143
6.1.4. Entorno multilingüe 143

Universitat d'Alacant
Universidad de Alicante

6.1.5. Arquitectura del sistema 146
6.1.6. Ejemplos de funcionamiento 146
6.1.7. Resultados 152
6.2. Otras aplicaciones 152
6.2.1. Prescripción de medicamentos 153
6.2.2. Selección de plantas medicinales 156
6.2.3. Ranking de temas relacionados en búsquedas médicas 156
6.2.4. Directorio médico 160

7. Conclusiones y trabajo futuro **163**
7.1. Conclusiones 163
7.2. Trabajo futuro 165
7.3. Publicaciones del autor 166
7.3.1. Algoritmo de clasificación de textos 166
7.3.2. Clasificación de artículos médicos 167
7.3.3. Ayuda para el diagnóstico médico y prescripción de medicamentos y plantas medicinales 168
7.3.4. Directorio médico utilizando artículos de la Wikipedia 169

A. Descriptores de categorías **171**
A.1. Reuters-21578, 90 categorías 171
A.2. Reuters-21578, 50 categorías con mayor número de muestras 173
A.3. OHSUMED, sub-árbol de enfermedades 174
A.4. OHSUMED, sub-árbol de enfermedades del corazón 175
A.5. OHSUMED, 50 categorías con mayor número de muestras 177

B. Palabras de parada **179**
B.1. Reuters-21578, 90 categorías 179
B.2. Reuters-21578, 50 categorías con mayor número de muestras 179
B.3. OHSUMED, sub-árbol de enfermedades 180
B.4. OHSUMED, sub-árbol de enfermedades del corazón 181

Universitat d'Alacant
Universidad de Alicante

B.5. OHSUMED, 50 categorías con mayor número de muestras 181
B.6. Diagnóstico médico 182

C. Influencia de los parámetros en el ranking de categorías **185**

D. Peso óptimo para las expresiones de los títulos **189**

E. Ejemplos adicionales de diagnóstico médico **191**
E.1. Ejemplo 3: Embolia pulmonar 191
E.2. Ejemplo 4: Ataxia 194

F. Consultas a la base de datos PubMed **197**

G. Aplicaciones Web: estadísticas y comparativas de uso **199**

Bibliografía **207**

Universitat d'Alacant
Universidad de Alicante

Índice de figuras

2.1. Aprendizaje inductivo 20

2.2. Representación gráfica del proceso de aprendizaje y clasificación automática de nuevos documentos 21

2.3. Representación de documentos 30

2.4. Selección de términos para reducir la dimensionalidad antes del proceso de aprendizaje. 34

2.5. Un árbol de decisión simple 39

2.6. Arquitectura de extracción de información basada en clasificadores 41

3.1. Ley de Zipf 54

3.2. Representación gráfica para el concepto de SVM 74

3.3. Línea de decisión para SVM con margen mínimo 74

4.1. Términos válidos con mayor cantidad de repeticiones. Corpus OHSUMED. . 95

4.2. Términos representativos de la categoría "Carcinoid Heart Disease" 99

5.1. Número de documentos para las 10 categorías más pobladas de Reuters-21578 109

5.2. Ejemplo de comportamiento para SVM variando el parámetro C 119

5.3. Deterioro en el rendimiento de clasificación binaria después de quitar cada parámetro 132

5.4. Deterioro en el rendimiento del ranking de categorías medido con AvgP después de quitar cada uno de los parámetros 134

5.5. Deterioro en el rendimiento del ranking de categorías medido con BEP después de quitar cada uno de los parámetros 135

Universitat d'Alacant
Universidad de Alicante

5.6. Deterioro en el rendimiento del ranking de categorías medido con maxF1 después de quitar cada uno de los parámetros 136

5.7. Deterioro en el rendimiento aplicando diferentes factores de ponderación para los títulos de los documentos. 137

6.1. Ejemplo de diagnóstico mediante ranking de categorías. 140

6.2. Ejemplo de diagnóstico después de la expansión de categorías de niveles superiores 144

6.3. Ejemplo de diagnóstico plano 145

6.4. Historial médico de paciente con quistes del sistema nervioso central 147

6.5. Ejemplo de diagnóstico para paciente con quistes del sistema nervioso central 149

6.6. Historial médico de paciente con angina inestable 150

6.7. Ejemplo de diagnóstico para paciente con angina inestable 151

6.8. Ejemplo de sugerencia en la prescripción para "dolor de cabeza" 154

6.9. Ejemplo de sugerencia en la prescripción para "rheumatoid arthritis" 155

6.10. Ejemplo de sugerencia de plantas medicinales para "insomnio". 157

6.11. Ejemplo de búsqueda de información sobre "allergies" en `searchmedica.com`. 157

6.12. Ejemplo de búsqueda de información sobre "allergies" en `lookformedical.com`.159

6.13. Ejemplo de recorrido del directorio web para localizar artículos relacionados con "Cardiovascular Diseases" 161

E.1. Historial médico de paciente con embolia pulmonar 192

E.2. Ejemplo de diagnóstico para paciente con embolia pulmonar 193

E.3. Historial médico de paciente con ataxia 195

E.4. Ejemplo de diagnóstico para paciente con ataxia 196

F.1. Consulta a la base de datos PubMed para recuperar artículos clasificados dentro de las categorías de principios activos 198

F.2. Consulta a la base de datos PubMed para recuperar artículos clasificados dentro de la categoría de plantas medicinales 198

F.3. Consulta a la base de datos PubMed para recuperar artículos clasificados dentro de cualquier categoría del árbol MeSH. 198

G.1. Comparativa de uso de `lookfordiagnosis.com`. 200

Universitat d'Alacant
Universidad de Alicante

G.2. Comparativa de uso de `lookfortherapy.com`. 201
G.3. Comparativa de uso de `lookformedical.com`. 202
G.4. Visitas de `lookfordiagnosis.com` por ubicación. 203
G.5. Visitas de `lookfortherapy.com` por ubicación. 204
G.6. Visitas de `lookformedical.com` por ubicación. 205

Universitat d'Alacant
Universidad de Alicante

Universitat d'Alacant
Universidad de Alicante

Índice de tablas

3.1. Comparación entre los dos modelos de Naive Bayes 77

3.2. Complejidad temporal de diversos clasificadores 85

3.3. Precisión medida con F1 entre distintos corpus y clasificadores ordenados por complejidad 85

5.1. Número de categorías y cantidad de documentos en el conjunto de entrenamiento y evaluación en los 5 primeros niveles del corpus OHSUMED, sub-árbol de enfermedades del corazón. 110

5.2. Cálculo de la precisión y cobertura micro y macro promediadas. 116

5.3. Ejemplo de cálculo de precisión mediante micro y macro promedios. 117

5.4. Valoraciones micropromediadas al utilizar diferentes características y ponderaciones sobre el corpus Reuters, 10 categorías con mayor número de muestras. 120

5.5. Valoraciones micropromediadas al utilizar diferentes características y ponderaciones sobre el corpus Reuters, 90 categorías. 121

5.6. Valoraciones micropromediadas al utilizar diferentes características y ponderaciones sobre el corpus OHSUMED, 23 categorías (sub-árbol de enfermedades). 121

5.7. Valoraciones micropromediadas al utilizar diferentes características y ponderaciones sobre el corpus OHSUMED, 102 categorías (sub-árbol de enfermedades del corazón). 121

5.8. Resultados utilizando BEP macro y micropromediado para Reuters, seleccionando las 10 categorías con mayor número de muestras. 122

5.9. BEP para OHSUMED, sub-árbol de enfermedades, seleccionando las 5 categorías con mayor número de muestras, y calculando también el valor micropromediado sobre el conjunto total de las 23 categorías. 123

5.10. F1 para Reuters, 90 categorías, y para OHSUMED, 102 categorías (sub-árbol de enfermedades del corazón). 123

Universitat d'Alacant
Universidad de Alicante

5.11. Comparación de resultados con los algoritmos Bayes, Rocchio, C4.5, kNN y SVM, utilizando el corpus Reuters, 90 categorías 125

5.12. Comparación de resultados con los algoritmos Rocchio, Perceptrón y MMP utilizando el corpus Reuters, 90 categorías 126

5.13. Comparación de resultados con los algoritmos AdaBoost.MH y AdaBoost.MR, utilizando el corpus Reuters, 50 categorías 126

5.14. Comparación de resultados con los algoritmos Bayes, Rocchio, C4.5, kNN y SVM, utilizando el corpus OHSUMED, 23 categorías (sub-árbol de enfermedades) 127

5.15. Comparación de resultados con los algoritmos BoosTexter.RV, Centroid Booster, y SVM light, utilizando el corpus OHSUMED, 102 categorías (sub-árbol de enfermedades del corazón) 128

5.16. Comparación de resultados con los algoritmos AdaBoost.MH y AdaBoost.MR, utilizando el corpus OHSUMED, artículos del año 1987, 50 categorías 128

5.17. Número de características en relación con F1 micropromediada 132

6.1. BEP obtenido al utilizar el clasificador para la obtención de diagnósticos médicos. 152

C.1. Resultados para Reuters después de aislar y quitar la influencia de cada uno de los parámetros del algoritmo. 186

C.2. Deterioro en el funcionamiento (%) para Reuters después de aislar y quitar la influencia de cada uno de los parámetros del algoritmo. 186

C.3. Resultados para OHSUMED (enfermedades del corazón) después de aislar y quitar la influencia de cada uno de los parámetros del algoritmo. 187

C.4. Deterioro en el funcionamiento (%) para OHSUMED (enfermedades del corazón) después de aislar y quitar la influencia de cada uno de los parámetros del algoritmo. 187

C.5. Resultados para OHSUMED (enfermedades) después de aislar y quitar la influencia de cada uno de los parámetros del algoritmo. 188

C.6. Deterioro en el funcionamiento (%) para OHSUMED (enfermedades) después de aislar y quitar la influencia de cada uno de los parámetros del algoritmo. . 188

D.1. Evaluación de los resultados utilizando Reuters y considerando diferentes factores de ponderación para las expresiones de los títulos 189

D.2. Deterioro en el funcionamiento (%) para Reuters aplicando diferentes factores de ponderación para las expresiones de los títulos. 190

Universitat d'Alacant
Universidad de Alicante

D.3. Evaluación de los resultados utilizando OHSUMED (enfermedades del corazón) y considerando diferentes factores de ponderación para las expresiones en los títulos 190

D.4. Deterioro en el rendimiento (%) para OHSUMED (enfermedades del corazón) aplicado diferentes factores de ponderación para las expresiones en los títulos. 190

D.5. Resultados para OHSUMED (enfermedades) considerando diferentes factores de ponderación para las expresiones en los títulos. 190

D.6. Deterioro en el funcionamiento (%) para OHSUMED (enfermedades) aplicando diferentes factores de ponderación para las expresiones en los títulos. . . . 190

Universitat d'Alacant
Universidad de Alicante

Universitat d'Alacant
Universidad de Alicante

Capítulo 1

Introducción

El rápido crecimiento de Internet está provocando que cambiemos nuestra forma de manejar información. Ya nos hemos habituado a trabajar con documentos en formato electrónico, desechando poco a poco el formato impreso. Hoy podemos buscar libros y noticias utilizando diversidad de soportes informáticos para acceder a gran variedad de páginas web, muchas de las cuales se encuentran accesibles sin coste alguno.

Cuando se dispone de mucha información, resulta conveniente clasificarla. Día a día, el número de documentos electrónicos se incrementa constantemente. Abogados, médicos, profesores, etc., todos necesitan de una herramienta que provea accesibilidad y simplicidad para clasificar y ordenar toda la información, de forma que se pueda localizar cualquier documento rápidamente. Después de un proceso de clasificación sobre un conjunto de textos, resulta más fácil encontrar información sobre un tema en concreto porque la búsqueda se limita a un conjunto más reducido de documentos cuyo contenido es similar.

Universitat d'Alacant
Universidad de Alicante

Si se trabaja con pocos documentos o sólo utilizamos información generada por nosotros, el proceso de clasificación manual puede resultar sencillo. Sin embargo, en la situación actual, donde todos podemos publicar en Internet contenidos textuales sobre cualquier tema, resulta imposible clasificarlos a mano. Por ello, es necesario utilizar un proceso automático para poder agrupar la información según diversos criterios. Además, dada la envergadura del problema, dicho proceso debería ser rápido, sin necesidad de supervisión y ajustable de manera automática al tipo de documentos sobre el que se aplique.

Aun consiguiendo este objetivo de organizar gran cantidad de documentos de manera eficiente, no todo son beneficios en la sociedad de la información. Al amparo de estas tecnologías han surgido nuevas aplicaciones, motivadas por el masivo incremento en la cantidad y variedad de información y su uso incontrolado. Un ejemplo cotidiano de sistemas de clasificación automática de documentos son los cada vez más necesarios filtros de páginas web y detectores de correos electrónicos no deseados (*spam*). Estos últimos clasifican todo el correo electrónico entrante y nos permiten descartar aquellos considerados por el sistema como no relevantes. Los filtros web, además, previenen que los menores accedan a contenido inapropiado, como violencia o pornografía, utilizando un proceso automático que cataloga ciertas páginas web

como inadecuadas.

Mirando hacia atrás, y analizando el desarrollo histórico de las tecnologías de clasificación de textos, encontramos que existen dos alternativas en el diseño de los clasificadores. Hasta la segunda mitad de los años 80 se optó por una solución basada en reglas generadas manualmente por expertos, una aproximación sencilla pero que requería un gran esfuerzo humano para crear las reglas y que exigía un diseño completamente nuevo para cada nuevo dominio sobre el que se quisiera aplicar el sistema. Sin embargo, en los años 90 la potencia de los ordenadores se incrementó notablemente y se hizo posible manejar grandes cantidades de documentos. Esto condujo al uso de soluciones basadas en *aprendizaje automático*, que se construían utilizando los siguientes elementos:

- Un corpus de ejemplos previamente etiquetados.
- Una taxonomía de clases posibles para asignar a los documentos.
- Un algoritmo de aprendizaje.
- Un conjunto de características extraídas de los documentos.

Este tipo de clasificadores aprende de forma inductiva partiendo de textos etiquetados previamente por expertos humanos, donde cada documento se representa por un conjunto de características que sirven de entrada para el algoritmo de aprendizaje. De esta forma, el clasificador es capaz de asignar una clase de la taxonomía definida a cada nuevo documento que llega sin etiquetar, basándose en la información previa proporcionada por el corpus de ejemplos. Esta opción se hizo muy popular por su gran precisión, por reducir considerablemente la intervención humana en su diseño y por permitir su utilización en diferentes dominios de manera sencilla, repitiendo simplemente el proceso de aprendizaje con otros documentos de entrada.

Desde entonces, diversos algoritmos de aprendizaje automático se han aplicado a la clasificación de textos. Entre ellos se pueden nombrar, por ejemplo, los *árboles de decisión* [Quinlan, 1986], *Naive Bayes* [Lewis y Ringuette, 1994] o los *k vecinos más cercanos* [Yang y Chute, 1994]. Sin embargo, pese a las grandes ventajas que aporta la aproximación basada en aprendizaje automático, estos algoritmos requieren una gran cantidad de textos durante el proceso de aprendizaje y por ello plantean dos grandes problemas:

- Cómo diseñar un algoritmo de aprendizaje utilizando de forma eficiente las características extraídas de cada documento, aplicando métodos o cálculos que ofrezcan una complejidad razonable.
- Cómo valorar y escoger un subconjunto adecuado de características para optimizar los resultados del proceso de clasificación.

Estos dos factores son interdependientes [Yang y Pedersen, 1997], de manera que los propios algoritmos de aprendizaje deberían ser capaces de determinar cuáles son las características más relevantes para la clasificación. Numerosos autores han revelado el efecto conseguido al utilizar un buen método de selección de características.

Dentro de la tarea de clasificación automática de textos, es habitual el uso de los propios términos del documento como características representativas del mismo durante la fase de aprendizaje y clasificación. A la hora de diseñar un sistema de este tipo podemos, por ejemplo, plantearnos qué relevancia pueden tener para identificar la categoría a la que pertenece un documento el uso de términos como "y" u "o", carentes de significado, pudiendo eliminarlos del conjunto de características empleado sin afectar al rendimiento del sistema. Podemos ir más allá y plantear que, una vez eliminadas dichas palabras sin contenido semántico, puede resultar interesante establecer un umbral de frecuencia de aparición a partir del cual se considera que un término determinado puede ser relevante o no para identificar la clase a la que pertenece [Joachims, 1998]. También, se puede realizar un estudio previo para establecer, por ejemplo, una cantidad máxima de términos a utilizar [Dumais et al., 1998]. Se intenta por tanto eliminar términos claramente irrelevantes y de entre los restantes, elegir o valorar de alguna forma los más adecuados.

Partiendo de los planteamientos expuestos, es normal que surjan ciertas preguntas: ¿cuánta influencia tiene la selección de características en el proceso de clasificación automática?, ¿hay una estrategia más adecuada que otra para escoger las características apropiadas?, ¿qué clasificador ofrece la mejor relación entre precisión de salida y consumo de recursos o tiempo de respuesta?. En la bibliografía podemos encontrar numerosas publicaciones que pretenden dar respuesta a estas preguntas [Di Nunzio y Micarelli, 2003; Mladenić et al., 2004; Debole y Sebastiani, 2004b]. Nuestra contribución para responder estos interrogantes consiste en una solución basada en conceptos no considerados hasta la fecha, así como su combinación con propuestas previas. Veremos que es posible obtener mejores métodos de selección y ponderación de características para incrementar el rendimiento de algoritmos existentes tales como SVM [Cortes y Vapnik, 1995]. Además demostraremos la flexibilidad adicional que ofrece nuestra propuesta con respecto a aproximaciones previas. Escoger ciertas características para representar cada categoría y utilizar una adecuada ponderación de las mismas, nos llevará a mejorar la calidad del proceso de aprendizaje y nos permitirá realizar una comparación directa entre los documentos a clasificar y cada una de las posibles categorías a asignar.

Las exigencias computacionales de ciertos algoritmos de aprendizaje a la hora de obtener resultados a gran escala, hacen que el ámbito de aplicación de éstos se reduzca. Desarrollar un clasificador más rápido, que consuma menos recursos, manteniendo o incluso mejorando los resultados a la hora de clasificar, abre la posibilidad a aplicaciones que de otra forma difícilmente podrían llevarse a la práctica. Por ejemplo, la base de datos PubMed[1] contiene más de 22 millones de referencias a artículos biomédicos y crece a un ritmo de 500.000 nuevas citas cada año. Aproximadamente el 15 % enlazan gratuitamente con el contenido completo

[1] http://www.ncbi.nlm.nih.gov/pubmed

de los textos, y todos ellos se encuentran ya clasificados en más de 25.000 categorías[2]. Sin embargo, aún teniendo fácil accesibilidad a los datos, no se encuentra en la literatura ninguna investigación que haya tratado de realizar pruebas con dicho corpus al completo.

En este sentido, este trabajo intenta afrontar nuevas aplicaciones de la clasificación de textos dentro del dominio médico, utilizando cientos de miles de artículos como datos de aprendizaje. Para ello, proporcionaremos demostraciones operativas de un clasificador de enfermedades en función de síntomas y de un buscador médico que permita refinar los resultados mediante sugerencias de términos similares o búsquedas jerárquicas. Además de la posibilidad de trabajar con grandes cantidades de documentos, mostraremos cómo nuestra aproximación es capaz de proporcionar una respuesta en tiempo real, independientemente del tamaño del corpus utilizado y sin que ello suponga una merma en la precisión obtenida.

En las siguientes secciones de este capítulo hablaremos de las necesidades que han motivado esta tesis y de los objetivos planteados, así como de la metodología seguida y la estructura de este trabajo.

1.1. Motivaciones

Durante los últimos años, la mayoría de los algoritmos de aprendizaje que han intentado abordar el problema de la clasificación de textos se han topado con una dificultad importante: el gran número de características que se utilizan para representar los documentos. En la mayoría de ocasiones, estas características son los propios términos (palabras o frases) que aparecen en los documentos. La cantidad de términos puede variar considerablemente dependiendo del tipo y el tamaño de la colección de textos utilizada, pero se puede hablar del orden de cientos de miles de términos para un corpus de tamaño moderado. Esto puede resultar prohibitivo para muchos algoritmos de clasificación [Yang y Pedersen, 1997], incluso con las tecnologías de computación actuales.

Debido a la creciente disponibilidad de documentos en formato digital, ha surgido la necesidad de diseñar y desarrollar técnicas especiales que permitan su procesamiento automático. Concretamente, los problemas que queremos afrontar y que motivan este trabajo son:

- La alta dimensionalidad de los espacios de características.
- La valoración de la relevancia de las características de aprendizaje.
- La aplicación de técnicas de aprendizaje y clasificación a tareas inabordables mediante algoritmos tradicionales.

Analizaremos a continuación las dificultades que presenta la clasificación de textos dentro de cada una de estas tres problemáticas.

[2] http://en.wikipedia.org/wiki/Medical_Subject_Headings y http://en.wikipedia.org/wiki/PubMed (Accedidas en diciembre de 2012).

1.1.1. El problema de la dimensionalidad

Trabajar con un número elevado de características puede dificultar la aplicación de muchos algoritmos de clasificación debido a su elevado coste computacional. En estos casos es recomendable reducir la cantidad de características mediante un procesado previo de los datos de entrada (documentos), o una selección o transformación de los mismos antes de ser utilizados. Pero ello puede tener su contrapartida, ya que una reducción agresiva nos puede dejar sin información suficiente para que el algoritmo induzca una solución válida. Sin embargo, un exceso de características, aunque técnicamente sean procesables, puede ocasionar la inclusión de mucho ruido que enmascare la información que realmente es importante para la clasificación. Así pues, hay que buscar un equilibrio entre el número de características de entrada seleccionadas y la calidad de los resultados obtenidos por el clasificador.

Establecer umbrales o pesos que permitan seleccionar las mejores características para un determinado dominio no es un proceso trivial, cuya dificultad se ve incrementada cuando se trata de afrontar de manera automática, ya que los valores obtenidos para un determinado dominio pueden no ofrecer resultados positivos al ser aplicados sobre otros conjuntos de datos. En el intento de automatizar la selección de características, muchos de los métodos incluyen la eliminación de términos no informativos utilizando valores estadísticos obtenidos del corpus [Yang y Wilbur, 1996; Peters y Koster, 2002]. Por ejemplo, existen ciertas palabras poco informativas a la hora de determinar la categoría a la que pertenece un texto, como los verbos auxiliares, conjunciones y artículos. Estas palabras se conocen como *palabras de parada* o *stopwords*. Existen numerosas listas con este tipo de términos que suelen emplearse para su eliminación del texto mediante un preproceso. Esto se lleva a cabo porque estas palabras aparecen en la mayoría de los documentos y en este sentido carecen de poder discriminatorio entre clases. Sin embargo, no dejan de ser listas prefijadas, idénticas para todos los clasificadores y todos los corpus.

En el extremo opuesto, otra forma de seleccionar características es eliminar aquellas palabras que casi no aparecen en ningún documento, considerándolas en algunos casos como errores ortográficos. Sin embargo, esta forma de proceder plantea una duda: es posible que una palabra mal escrita no sea tal, sino una variación específica del corpus en cuestión (por ejemplo, siglas o una abreviatura). En este caso, ¿no sería mejor valorarla más, pues es muy peculiar y significativa?.

El reemplazo de cada palabra por su raíz o lema es también otra técnica muy común en el preproceso de documentos para la reducción de características [Caropreso et al., 2001; Gaustad et al., 2002; Granitzer, 2003]. Para reducir el tamaño del conjunto inicial de características, se consideran como el mismo término todas aquellas palabras que tienen la misma raíz o lema. Por ejemplo, en el caso del lema, las palabras "entrenar", "entrenamiento" o "entrenador" se reducen a la forma común "entrenar". A pesar de la amplia difusión de tales métodos, existen ciertas dudas sobre la eficiencia de una reducción agresiva como las realizadas por el algoritmo de Porter [Braschler y Ripplinger, 2004], ya que en estos casos tanto verbos como sustantivos quedan muchas veces reducidos a la misma raíz o lema. Nosotros proponemos ser más selectivos en este sentido, reduciendo sólo los plurales a su forma singular

y los femeninos a masculino, manteniendo intacta la distinción entre verbos y sustantivos a nivel morfológico.

1.1.2. Relevancia de los términos

La asignación de un peso a cada característica es otro punto a abordar, ya que permite asignar más o menos importancia a una característica con respecto al resto a la hora de realizar la clasificación. A menudo, para representar cada documento a partir de una lista de términos, se utiliza un valor booleano que indica si éste aparece o no en el documento. Otras soluciones posibles consisten en incluir el número de veces que el término aparece en el documento, o incluir la frecuencia de aparición de los términos normalizada por la longitud del documento (número de términos que contiene). Esto último resulta interesante en situaciones donde la longitud entre los documentos de un corpus varía significativamente. Por otra parte, en documentos pequeños las palabras no suelen repetirse y el indicador booleano puede resultar tan informativo como el empleo de medidas de frecuencia, desembocando en una simplificación importante del problema.

Basándose en estas consideraciones se han desarrollado numerosos métodos de selección y ponderación de características. Una de las variantes más conocidas a la hora de representar un documento es la bolsa de palabras (*Bag of Words - BOW*). En este modelo de representación, un texto se representa como una colección no ordenada de palabras, sin tener en cuenta ningún tipo de información gramatical, ni tan siquiera el orden original de los términos. En este modelo de representación, es habitual ponderar la relevancia de los términos en el corpus teniendo en cuenta su frecuencia de aparición (*Term Frequency - TF*) y el número de documentos en los que aparecen (*Inverse Document Frequency - IDF*). Es lo que se conoce como modelo *TF-IDF*. Sin embargo, este valor no considera la distribución de los términos entre las diferentes categorías posibles. Por ejemplo, la mayoría de las propuestas actuales considerarían muy poco relevante un término que apareciera en un solo documento perteneciente a una sola categoría y probablemente lo descartarían. ¿ Y si este término que estamos descartando es incluso más valioso que el resto?. Igualmente podríamos situarnos en el extremo opuesto, ya que también es posible que no se valoren adecuadamente aquellos términos poco discriminativos por su representatividad en muchas categorías.

1.1.3. La clasificación de textos en el mundo real

Muchas investigaciones dentro del campo de la clasificación de textos se han volcado en intentar mejorar la precisión y la cobertura obtenida por los sistemas, tratando de reducir el coste computacional y los recursos requeridos tanto en la fase de aprendizaje como en la clasificación, con el objetivo final de ampliar el ámbito de aplicación en el mundo real. Por

ejemplo, se ha intentado mejorar el uso de BOW mediante *stemming*[3], *n-gramas*[4], selección de características, sintagmas nominales, metadatos, normalización de pesos, etc. Sin embargo, los resultados obtenidos en las múltiples investigaciones realizadas en este campo no son del todo coherentes. En algunos experimentos el uso de frases[5] ocasionaba un rendimiento menor del sistema [Lewis, 1992a,b; Apté et al., 1994b]. A pesar de estos resultados, la investigación en el uso de características más sofisticadas no se detuvo [Schütze et al., 1995; Mladenić y Grobelnik, 1998]. Incluso para una persona ajena a cualquier investigación, puede resultar intuitivo pensar que frases como "gripe A" o "gripe estacional" pueden resultar mucho más informativas a la hora de identificar el tema sobre el que trata un texto que la aparición separada de los términos "gripe", "A" y "estacional". Sin embargo, en la actualidad la mayoría de las técnicas de clasificación de textos siguen basándose en el uso de términos individuales.

Pese a que el uso de frases tiende a reducir la ambigüedad de los términos, incrementa notablemente el número de características del problema y crea un problema de dispersión, ya que muchas de estas frases aparecen en muy pocos documentos. Por ejemplo, utilizando frases de dos palabras el número de términos se podría llegar a elevar al cuadrado. Otro problema del uso de frases es el incremento de la redundancia por la alta similitud de los términos. Esto es debido a que, si para cada palabra que compone una frase existe un número determinado de sinónimos, al considerar dicha frase como un término existirán otros muchos términos similares. Por ejemplo, si consideramos "plantas medicinales" como un único término, nos podemos encontrar con expresiones equivalentes como "hierbas medicinales", "hierbas curativas", "plantas terapéuticas", "plantas curativas", etc., que tienen el mismo significado y que por lo tanto no consiguen incrementar la riqueza del vocabulario.

Otro planteamiento problemático en el que se basan modelos de representación más complejos que BOW, es que si dos frases tienen el mismo significado aparecerán en documentos que traten del mismo tema. Sin embargo, en numerosas ocasiones, en los textos escritos en lenguaje natural ocurre lo contrario. Podemos utilizar como ejemplo las expresiones "vías respiratorias" y "vías aéreas". Ambas tienen el mismo significado dentro del ámbito médico, pero la segunda presenta el inconveniente de aparecer también en documentos relacionados con la aeronáutica.

Desde el punto de vista de nuestra investigación, puede resultar muy interesante valorar las diferentes combinaciones de todos los factores expuestos anteriormente para la obtención de una clasificación más precisa de los documentos. Además, consideramos muy relevante el procesado de los textos de entrenamiento para la obtención de algunos datos estadísticos previos a la construcción del clasificador. Dicho proceso puede simplificar en gran medida el algoritmo de clasificación, manteniendo un alto nivel de precisión y mejorando considerablemente el tiempo de respuesta, lo cual es un aspecto fundamental para conseguir ampliar el ámbito de aplicación de estos sistemas.

[3] En lingüística, *stemming* es el proceso de reducir un término a su raíz.

[4] El término *n-grama* se usa para designar a los fragmentos de una secuencia más grande hecha de palabras o letras.

[5] Entendemos como *frase* un caso particular de n-grama formado por dos o más palabras consecutivas.

1.2. Objetivos

Después de exponer las dificultades que pretendemos abordar, podemos resumir brevemente nuestro objetivo: desarrollar un sistema de clasificación de textos que se adapte fácilmente a diferentes corpus, seleccionando automáticamente de cada documento la información necesaria para obtener una representación estándar lo más descriptiva y discriminativa posible. Mediante este planteamiento pretendemos, en primer lugar, poder generar nuestra propia representación de los documentos de entrada que sirva para alimentar a algoritmos de clasificación ya existentes. En segundo lugar, generar una representación adecuada del corpus nos va a permitir mejorar los indicadores de pertenencia de las muestras de entrenamiento a cada posible categoría, consiguiendo simplificar el proceso de clasificación de nuevos documentos y permitiendo tomar decisiones más rápidamente y con la necesidad de menos recursos.

Este objetivo principal se puede desglosar en una serie de objetivos intermedios:

- Seleccionar las expresiones adecuadas para representar cada categoría.
- Valorar correctamente cada expresión, independientemente del tipo y tamaño de los textos utilizados.
- Seleccionar automáticamente una lista de expresiones no relevantes específica de cada corpus, comparable a la que elaboraría un equipo de expertos humanos utilizando conocimientos del dominio.
- Evaluar la contribución individual de cada una de las aproximaciones utilizadas para demostrar su validez dentro del proceso global.
- Establecer un método sencillo de comparación de pertenencia de un documento a cada categoría utilizando datos estadísticos, sin necesidad de utilizar otros algoritmos de clasificación.
- Comparar los resultados obtenidos utilizando diferentes corpus, que se distingan entre sí por las siguientes características:
  - Temáticas diferentes.
  - Distribuciones dispares en la cantidad de documentos de entrenamiento de cada categoría.
  - Diversos tamaños de documentos.
  - Tamaño del corpus, tanto en número de documentos de entrenamiento como en número de posibles categorías.
- Procesar corpus de gran tamaño para aplicar nuestra aproximación a problemas del mundo real.

Otros objetivos adicionales que trataremos de abordar son:

- Aplicación en el ámbito médico.
- Desarrollo de aplicaciones con respuesta en tiempo real.
- Evaluación del uso de dichas aplicaciones.

En resumen, el objetivo perseguido con esta investigación es obtener en primer lugar una representación óptima de los documentos que pueda ser utilizada por algoritmos tales como SVM, seleccionando y ponderando las diversas características, tanto durante el proceso de entrenamiento como en la fase de clasificación. En segundo lugar trataremos de realizar nuestra propia propuesta para establecer el grado de pertenencia de un nuevo documento a las diversas clases de un corpus, utilizando la base conceptual del aprendizaje automático.

1.3. Metodología

Para la consecución de los objetivos mencionados en la sección anterior utilizaremos un serie de consideraciones específicas, llevando a cabo una metodología consistente en cinco pasos:

1. **Procesado de los documentos de entrenamiento**. Se analizarán documentos ya clasificados para obtener una representación más adecuada de los mismos, de manera que pueda ser utilizada para alimentar el proceso de aprendizaje del clasificador. A grandes rasgos, los pasos a seguir serán:
   - Estudiar qué términos pueden llegar a contener más información.
   - Determinar qué características establecen la popularidad de una expresión.
   - Contabilizar estadísticas y frecuencias de aparición que nos ayuden a establecer relaciones entre expresiones y categorías.
   - Descartar expresiones poco representativas, eliminando datos innecesarios.
2. **Validación de los datos para el aprendizaje**. Se utilizarán las características y pesos obtenidos en el paso anterior para entrenar un conocido algoritmo de clasificación (SVM) y se evaluará a continuación el proceso de clasificación de nuevos documentos. Aunque el uso de este algoritmo no es necesario en nuestro sistema, sí resultará muy útil emplearlo como algoritmo base del que ya disponemos de cifras, para establecer comparativas que nos permitan validar la calidad de la representación de los datos de entrada.
3. **Clasificación de nuevos documentos mediante suma de pesos**. Se planteará un algoritmo de clasificación propio, consistente en extraer las expresiones relevantes de los documentos a clasificar para compararlas con las expresiones relevantes de cada

Universitat d'Alacant
Universidad de Alicante

categoría, calculando la suma de los valores de las expresiones coincidentes entre documento y categoría. De esta forma observaremos cómo se puede obtener una lista ordenada realizando una simple valoración de pertenencia a cada categoría.

4. **Evaluación**. Se utilizarán dos corpus con características muy diferentes (Reuters y OHSUMED), que a su vez se dividirán en diversos subconjuntos para poder comparar los resultados con experimentos realizados por otros investigadores. También se evaluarán los resultados mediante las diversas combinaciones de cada uno de los parámetros establecidos durante todo el proceso.

5. **Aplicación**. Se desarrollarán cuatro aplicaciones diferentes mediante la metodología de clasificación propuesta, utilizando cientos de miles de artículos médicos en el proceso. Dichas aplicaciones estarán accesibles a través de Internet para el público en general:

   - Clasificación de enfermedades a partir de historias clínicas.
   - Clasificación de plantas medicinales y principios activos en función de síntomas.
   - Clasificación de términos relacionados con palabras clave de búsquedas médicas.
   - Clasificación jerárquica de artículos de la Wikipedia para la creación de un directorio médico.

   Un aspecto a dejar claro sobre las aplicaciones propuestas es que los temas relacionados con la medicina son muy extensos y de muy diferente índole. Por ejemplo, en la literatura científica las técnicas y herramientas empleadas para el procesamiento de artículos sobre proteínas no es igual al que se le da a los documentos que tratan sobre enfermedades. Por esta razón, la información que se expondrá en las siguientes secciones no estará enfocada a medicamentos o enfermedades, sino que se tratará de presentar un método general que sea independiente del tema médico específico con el que se relacione.

Universidad de Alicante

Es importante mencionar que, en los trabajos previos realizados en el área de clasificación de textos médicos, los experimentos realizados se hacen sobre el resumen de cada uno de los documentos tratados y no sobre el texto completo [Yang y Pedersen, 1997; Yang y Liu, 1999; Yang, 1999; Ceci y Malerba, 2007]. Algunos de los autores argumentan que deben limitar sus experimentos al no tener todos los documentos su contenido disponible libremente en la Web, mientras que otros dicen que el resumen proporciona la base de conocimiento suficiente para hacer una buena clasificación. Aunque para algunos de nuestros propósitos disponemos de gran cantidad de artículos de libre distribución (casi dos millones), nosotros utilizaremos también sólo los resúmenes, ya que este criterio nos permitirá realizar comparaciones con otras investigaciones realizadas previamente.

1.4. Estructura de la tesis

A continuación se describe brevemente el contenido de cada uno de los capítulos que siguen a esta introducción:

- En el **capítulo 2** expondremos la teoría básica que nos permitirá entender el proceso general que se sigue para la construcción de un clasificador de textos.
- En el **capítulo 3** realizaremos un recorrido de los trabajos previos en el área, exponiendo algunas de las soluciones propuestas hasta la fecha para abordar el problema de la clasificación textual.
- En el **capítulo 4** enumeraremos los conceptos que fundamentan las nuevas aportaciones que realizamos, justificando y razonando cada planteamiento. A continuación describiremos detalladamente el nuevo algoritmo propuesto. Indicaremos con exactitud los pasos a seguir para completar el procesado de los textos y la extracción, ponderación y selección de características.
- En el **capítulo 5** demostraremos la validez del algoritmo planteado mediante pruebas exhaustivas que compararán los resultados obtenidos en investigaciones previas con los obtenidos en esta tesis. A su vez, comprobaremos la contribución individual de cada uno de los componentes de nuestro sistema.
- En el **capítulo 6** mostraremos un nuevo campo de aplicación para la clasificación de textos. Aprovechando la escalabilidad del algoritmo propuesto, propondremos una solución para obtener sugerencias de diagnóstico médico partiendo del texto plano contenido en publicaciones científicas y en historiales de pacientes. Para demostrar la validez de los resultados, llevaremos a cabo su evaluación utilizando casos reales. Además propondremos otras aplicaciones dentro del campo de la medicina: prescripción de medicamentos, prescripción de plantas medicinales, buscador de términos médicos, y directorio médico. Se proporcionará una versión operativa en Internet para acceder a las mismas.
- En el **capítulo 7** plantearemos nuestras conclusiones sobre la utilidad de las aportaciones realizadas y su corroboración después de las evaluaciones llevadas a cabo. Además se enumerarán las publicaciones del autor, y el trabajo futuro previsto dentro de esta línea de investigación.
- En los **apéndices A y B** se muestra el listado de los descriptores de categorías para los conjuntos de prueba utilizados y las listas de expresiones vacías obtenidas durante el proceso de aprendizaje.
- En los **apéndices C y D** se muestran mediciones de la calidad de los resultados después de aplicar diversos ajustes en cada uno de los parámetros del sistema para observar su efecto sobre el ranking de categorías devuelto por el algoritmo de clasificación.
- En el **apéndice E** podemos encontrar un par de ejemplos adicionales para observar el funcionamiento del algoritmo de clasificación dentro del ámbito del diagnóstico médico partiendo de dos historias clínicas reales.
- En el **apéndice F** observamos consultas realizadas a la base de datos PubMed para obtener el texto de diversos artículos médicos ya clasificados.

Universitat d'Alacant
Universidad de Alicante

- En el **apéndice G** incluimos gráficos de estadísticas de uso de las páginas web que contienen las nuevas aplicaciones propuestas. Además, se presentan comparativas con páginas de tamaño y contenido similares.

Universitat d'Alacant
Universidad de Alicante

Capítulo 2

La clasificación de textos

Como se define en [Lewis y Gale, 1994], la *clasificación de textos*[1] es la agrupación automática de entidades textuales en una o más clases predefinidas, considerando así cada clase como una categoría semántica que aglutina documentos que tienen ciertas propiedades en común. Por ejemplo, si leemos varias noticias sobre diferentes temáticas, tales como deportes, política, entretenimiento, etc., observaremos que los textos de cada sección nos resultan similares. Los sistemas de clasificación son capaces de analizar nuevos documentos para captar automáticamente dichas semejanzas, y pueden en consecuencia decidir de qué trata una nueva noticia que se reciba como entrada.

Para conseguir clasificar documentos nuevos primero debemos establecer una serie de condiciones de pertenencia para cada categoría, las cuales muchas veces se deducen mediante un proceso de aprendizaje basado en ejemplos ya clasificados previamente. Después, basta con evaluar dichas condiciones para decidir cómo etiquetar los nuevos datos de entrada. En otras palabras, conseguiremos categorizar un documento si distinguimos las características que lo hacen distinto de otros, lo que nos permitirá relacionarlo con ciertas categorías. Una vez logrado esto, un documento podrá situarse en varias clases, sólo una, o en ninguna. Por ejemplo, si tratamos de filtrar textos considerándolos como relevantes o irrelevantes, se asume que a cada uno se le asigna exactamente una clase.

En términos formales, existe una función T definida en $(D \times C)$ tal que $T(d_i) = c_i$ donde D es el conjunto de los documentos disponibles, C es el conjunto de posibles categorías, d_i es un documento cualquiera y c_i es el conjunto de las categorías a las que pertenece el documento d_i.

Por lo tanto, el objetivo será encontrar una función T' que se parezca lo máximo posible a la función T ya definida. Tal definición y coincidencia establecerá la efectividad del proceso

[1] También conocida como categorización de textos. Se debe clarificar que dentro del ámbito del aprendizaje automático o de la recuperación de información, el término "clasificación" es más genérico, pues se refiere comúnmente a procesos que agrupan entidades (se puede consultar [Lewis, 1992b] para obtener más detalles). Nosotros sin embargo consideraremos las expresiones "clasificación de textos" y "categorización de textos" como sinónimos.

de clasificación automática. En este punto conviene aclarar que la obtención de T' no es un proceso trivial, y que los diversos pasos y métodos han evolucionado mucho durante los últimos años. Además, para resolver el problema que planteamos, debemos entender un texto como un documento compuesto de palabras. Esta consideración es importante dado que en la actualidad se están dirigiendo muchos esfuerzos para intentar clasificar documentos que contienen audio, imágenes y vídeo, desembocando en un problema de interpretación y representación que no se abordará en esta tesis. En el caso de un texto, si bien podría ser interpretado de distintas formas, estas formas están más acotadas porque las palabras cuentan con una definición establecida y común para todos.

2.1. Tipos de clasificación

Dado un documento cualquiera, el objetivo de un clasificador es asignarle una o varias categorías con las que se supone que está relacionado. Esto se puede reflejar de formas diferentes, dependiendo de la cantidad de clases que se puedan asignar a un documento, o del uso que se pretenda dar al clasificador, es decir, si se espera una respuesta en tiempo real o si por el contrario, se pretende utilizar directamente el resultado obtenido sin intervención alguna del usuario.

2.1.1. Clasificación binaria vs. clasificación multiclase

Por un lado nos podemos encontrar con ciertas situaciones en el mundo real que requieran de un clasificador que decida simplemente si un determinado documento pertenece a una categoría o no. Este tipo de clasificación se conoce como *simple* o *binaria* (por la decisión booleana que se aplica). Podemos citar por ejemplo el caso del filtro de correos electrónicos, donde un clasificador determina si un correo debe ser considerado como spam o no.

En muchos otros casos, los problemas que nos encontramos dependen de múltiples categorías. Es el caso por ejemplo de la clasificación de publicaciones científicas en el ámbito médico, donde un mismo documento puede tratar de varios temas, es decir, puede pertenecer a una o varias clases (*clasificación multiclase*). Por ejemplo, un texto que hable sobre el tratamiento de la miopía con láser, podría etiquetarse como relacionado con "Oftalmopatías" y con "Queratomileusis por Láser In Situ".

2.1.2. Pivote en el documento vs. pivote en la categoría

A la hora de diseñar el sistema de clasificación, debemos considerar que posiblemente no dispongamos desde el principio de todos los nuevos documentos que se pretenden clasificar. Para aplicaciones que requieran trabajar en tiempo real se recomienda realizar la clasificación utilizando *pivote en el documento*, pues nos interesa encontrar todas las categorías que puedan

estar relacionadas con un determinado documento que nos acaba de llegar. Por ejemplo, en el caso del diagnóstico médico, el documento a clasificar sería la lista actualizada de síntomas de un paciente, y el clasificador deberá proporcionar un listado de todas las posibles enfermedades con las que puedan estar relacionados dichos síntomas.

Sin embargo, si disponemos de todos los documentos, pero existe la posibilidad de que surjan nuevas categorías, resulta más conveniente utilizar *pivote en la categoría*, pues en este caso se trata de encontrar todos los documentos que puedan estar relacionados con un determinado tema. Este método es el recomendado por ejemplo para la clasificación jerárquica de páginas web, pues es común que surjan nuevas categorías y se eliminen otras, o simplemente cambie parcialmente la estructura de la taxonomía que se esté utilizando.

2.1.3. Asignación directa vs. ranking de categorías

La *asignación directa* de categorías se suele utilizar en sistemas completamente automáticos que no requieren de la intervención del usuario, al menos para la toma de decisiones de clasificación. Para llevarla a cabo, la mayor parte de los modelos existentes deciden sobre la pertenencia de un documento a una categoría eligiendo entre dos valores: verdadero (el documento pertenece a la categoría) o falso (el documento no pertenece a la categoría).

Si el clasificador mantiene una relación de interactividad con el usuario, puede resultar también deseable obtener la lista de posibles clases ordenadas por relevancia (*ranking de categorías*). Esto es de mayor utilidad en los casos en los que el proceso de clasificación no es realmente automático, sino asistido o semiautomático. De esta forma, la decisión final de clasificar un documento en una categoría cae directamente en manos del usuario, pues el clasificador le proporciona las categorías supuestamente más relevantes y le permite escoger las más adecuadas según su propio criterio.

Universidad de Alicante

La simple información de pertenencia o no pertenencia de cada documento a cada clase resulta insuficiente para establecer un ranking. El clasificador no sólo deberá proporcionar las diversas clases con las que se encuentra relacionado un documento, sino que además deberá aportar una noción del grado de pertenencia. Por ejemplo, un documento puede pertenecer a una clase en un grado que va de 0 a 1, siendo 0 el grado de exclusión máximo (el documento no trata en absoluto de la materia identificada para esta clase) y 1 el grado de pertenencia máximo (tratando el documento netamente sobre el tema de la clase en cuestión). Así ya estaríamos hablando de un clasificador que proporciona ranking de categorías, puesto que el resultado se puede convertir en una lista ordenada de etiquetas utilizando los números reales que reflejan las valoraciones que hacemos de los documentos respecto de las clases.

Más formalmente, la asignación directa se puede definir como una función $T \rightarrow \{D, C\}$, donde $T(d_i, c_i) = \{V, F\}$, donde el valor V representa pertenencia a la categoría y F representa la no pertenencia. Por otra parte, utilizando la definición de [Sebastiani, 1999], el ranking de categorías puede ser entendido como la tarea de asignar un valor $\{1, \cdots, m\}$ para cada una de las entradas de la siguiente matriz de decisión:

	d_1	d_2	$\cdots$	d_j	$\cdots$	d_n
c_1	a_{11}	a_{12}	$\cdots$	a_{1j}	$\cdots$	a_{1n}
c_2	a_{21}	a_{22}	$\cdots$	a_{2j}	$\cdots$	a_{2n}
$\cdots$	$\cdots$	$\cdots$	$\cdots$	$\cdots$	$\cdots$	$\cdots$
c_i	a_{i1}	a_{i2}	$\cdots$	a_{ij}	$\cdots$	a_{in}
$\cdots$	$\cdots$	$\cdots$	$\cdots$	$\cdots$	$\cdots$	$\cdots$
c_m	a_{m1}	a_{m2}	$\cdots$	a_{mj}	$\cdots$	a_{mn}

donde $C = \{c_1, c_2, \cdots, c_m\}$ es un conjunto de categorías predefinidas y $D = \{d_1, d_2, \cdots, d_n\}$ es un conjunto de documentos para ser categorizados. En este caso, un valor de 1 en a_{ij} se interpreta como una decisión de categorizar d_j en c_i como el caso más probable, mientras que un valor de m se interpreta como una decisión de categorizar d_j en c_i con la probabilidad más baja.

2.2. Aproximaciones al diseño de clasificadores

El interés por la categorización de textos surgió hace varias décadas. Ya en 1960 se empezaron a dar los primeros pasos en esta área. Por ese entonces, debido a las limitaciones de los primeros ordenadores, el diseño de los clasificadores tendía más hacia una vertiente manual, donde se dependía en gran medida del factor humano para el correcto diseño del clasificador.

Las primeras soluciones requerían de la ayuda de expertos en el área específica del ámbito de aplicación. Éstos analizaban los documentos que se pretendían clasificar para extraer todas aquellas peculiaridades que caracterizaban especialmente la temática de los mismos. Además se necesitaba desarrollar un software específico que extrajera dichas características de nuevos documentos y evaluara todos los factores que se habían definido manualmente. Esto planteaba lógicamente grandes inconvenientes, pues los costes y el tiempo de producción eran proporcionales al número de clasificadores que se pretendieran construir, ya que cada uno se diseñaba específicamente para el ámbito de aplicación donde se fuera a utilizar.

Con la aparición de las nuevas generaciones de ordenadores, se incrementó considerablemente la capacidad de procesamiento. Esto permitió a numerosos investigadores prescindir en gran medida del factor humano, avanzando hacia un proceso de automatización en la construcción de los clasificadores y dando lugar a otro tipo de procedimientos. Comenzaron a desarrollarse programas que permitían procesar una gran cantidad de datos de entrenamiento, retenían cierta información y a continuación eran capaces de procesar y clasificar nuevos documentos sin intervención humana.

De esta forma, podemos agrupar los sistemas de clasificación de textos en dos grandes familias, dependiendo de la aproximación que se utilice: basada en conocimiento o basada en aprendizaje automático. Describiremos ambas con más detalle a continuación.

2.2.1. Sistemas basados en conocimiento

Los primeros métodos que intentaron abordar la categorización de textos se basaban en conjuntos de reglas. Tal concepto se conoce ahora como *ingeniería del conocimiento* (*knowledge engineering*). Las reglas eran instrucciones condicionales del siguiente estilo:

Si $< condición_i >$ **Entonces** $< clase_j >$.

En [Castillo et al., 1996] podemos encontrar diversos ejemplos reales de sistemas basados en reglas. De entre ellos mostramos el conjunto de reglas para sacar dinero de un cajero automático:

- **Si** tarjeta caducada **Entonces** pago no autorizado
- **Si** NIP incorrecto **Entonces** pago no autorizado
- **Si** intentos excedidos **Entonces** pago no autorizado
- **Si** saldo insuficiente **Entonces** pago no autorizado
- **Si** límite excedido **Entonces** pago no autorizado

Actualmente todavía se puede encontrar software basado en esta filosofía, aunque algo más refinado, como es el caso de Spamassassin[2], un detector de correo no deseado (*spam*) que basa sus decisiones en una serie de condiciones preestablecidas, como por ejemplo si se trata de correo HTML, si el remitente tiene números en su composición, si existe el dominio de origen, si contiene ciertas palabras clave, etc. A cada una de estas características se le asigna una ponderación, de manera que si al evaluarlas todas el correo en cuestión sobrepasa un determinado umbral, es clasificado como spam.

Esta aproximación tiene una serie de inconvenientes. El primero de ellos se refiere a la construcción de las condiciones, pues para ello se requiere un experto en el área de investigación. Así, por ejemplo, en el caso de Spamassassin, son muchos los individuos que voluntariamente han aportado su experiencia para la confección de las reglas. Sin embargo, el caso de este software no es la norma. En la mayor parte de las áreas de investigación, encontrar expertos en el dominio de la aplicación es difícil, además de que los costes relacionados con dicho asesoramiento suelen ser muy elevados.

Un segundo inconveniente viene dado por el desarrollo del software. No sólo se requiere de un experto que especifique las reglas, sino también de quien interprete esas reglas y construya el clasificador utilizando un lenguaje computacional.

El tercer inconveniente tiene que ver con la portabilidad de la solución. Como estos categorizadores son construidos para una problemática particular, cambiarlos de un dominio de aplicación a otro puede llegar a ser casi imposible. También nos enfrentaremos a una situación

[2] http://spamassassin.apache.org.

similar si queremos incorporar una nueva categoría. En tales casos, probablemente resultará más fácil reconstruir completamente el clasificador.

Por último, nos encontramos con la parametrización, que también debe ser realizada por alguien con conocimientos del área. Por ejemplo, en el caso de Spamassassin el porcentaje que aporta cada regla a la evaluación final del correo, así como el umbral a partir del cual un correo es considerado spam, dependen de cada usuario. Las últimas versiones permiten ajustar estos parámetros de forma dinámica indicando manualmente si un determinado correo es o no spam, permitiendo así que la misma aplicación modifique los umbrales basándose en esta experiencia.

Pese a estos inconvenientes, en la actualidad no se puede descartar completamente el construir un clasificador mediante técnicas de ingeniería del conocimiento, porque se ha demostrado la alta efectividad de estos una vez ajustados. Por ejemplo, el clasificador de noticias Construe [Hayes y Weinstein, 1990], después de muchos años afinando reglas y parámetros, consiguió una efectividad por encima del 90 %. Aunque tal como se afirma en [Sebastiani, 2002], no se encuentran otras publicaciones que confirmen dichos resultados, sí que existen testimonios que avalan el uso de reglas en aplicaciones reales tales como el mencionado control de spam.[3] En cualquier caso, este tipo de aproximación está pasando a ser cada vez menos habitual, y aún resultando interesante conocer la evolución histórica de unas soluciones que todavía perduran, no profundizaremos más en dichos métodos.

2.2.2. Sistemas basados en aprendizaje automático

La otra modalidad de categorización de textos es el *aprendizaje automático* (*machine learning*). Viene haciéndose popular desde la década de 1980, y por estos días es, sin lugar a duda, lo opción más utilizada. Por ello, y por las motivaciones descritas en la sección 1.1, nosotros también centraremos nuestros esfuerzos en este ámbito y de ahora en adelante, dedicaremos toda esta tesis a procedimientos relacionados con aprendizaje automático.

En esta modalidad ya no hay expertos que dicten reglas. Se dispone de un módulo que aprende de manera inductiva a partir de ejemplos preclasificados, teniendo en cuenta si un documento de texto pertenece a determinada categoría o no. Es lo que se conoce como aprendizaje supervisado [Debole y Sebastiani, 2003], pues los resultados se obtienen mediante un algoritmo que es entrenado sólo con información que fue clasificada (supervisada) previamente. Debemos tener en cuenta que no siempre disponemos de todos los documentos ya agrupados por categorías. Por ello, podemos encontrar tres variantes del proceso de aprendizaje [Zhu, 2005; Tomás, 2009]:

- Supervisado. Todos los ejemplos del corpus están etiquetados y sólo debemos encontrar una función que relacione un conjunto de entradas con un conjunto de salidas.

[3] Se puede observar por ejemplo el uso de filtros para el control de spam en diversas universidades [Chile, 2010; Valencia, 2012].

- No supervisado. Se analizan los datos de entrada sin clasificar para intentar descubrir patrones comunes que permitan separar los ejemplos en grupos (*clusters*).

- Semisupervisado. Se utilizan dos conjuntos: uno que contiene textos clasificados y otro del que no disponemos de dicha clasificación. El objetivo principal en la mayoría de los casos consiste en aprender del primer conjunto para conseguir etiquetar a continuación el resto de documentos no clasificados. Una vez completado este proceso, ambos conjuntos se pueden utilizar como datos de entrada, de forma similar al método supervisado.

En la mayoría de los clasificadores, el aprendizaje automático se basa en la información intrínseca del documento, despreciando información externa como su origen, autor, formato, etc. Sin embargo, en [Sebastiani, 2002] se justifica que estos datos también deben ser considerados, puesto que aportan importante información respecto de la categoría a la cual debiera ser asignado y le proporcionan al clasificador mayor objetividad.

Más formalmente, para cada categoría c_i, se suministra al algoritmo de aprendizaje una serie de características extraídas automáticamente de un conjunto de documentos D previamente clasificados como pertenecientes a c_i. Con esta filosofía, si se actualiza el conjunto original de categorías, o si el sistema se porta a un dominio completamente diferente, sólo será necesaria la construcción automática e inductiva de un nuevo clasificador utilizando un conjunto diferente de documentos ya clasificados. Además, se simplifica considerablemente el proceso del diseño, pues permite ver la construcción de un clasificador para el conjunto de categorías $C = \{c_1, c_2, \cdots, c_m\}$ como m tareas independientes que se pueden abordar por separado. Puesto que es más fácil clasificar documentos que construir y afinar reglas de clasificación, en general resulta más rápido construir y poner en funcionamiento clasificadores basados en aprendizaje automático que basados en conocimiento.

De este modo, de acuerdo con la hipótesis del aprendizaje inductivo, cualquier clasificador que consiga aproximar la función objetivo sobre un conjunto suficientemente grande de documentos ya clasificados, también será capaz de aproximar la función objetivo sobre otro conjunto de documentos inéditos [Mitchell, 1997]. En otras palabras, el clasificador se podrá utilizar para predecir el etiquetado de clases de nuevos documentos no vistos ni procesados anteriormente.

Mediante la representación gráfica que se muestra en la figura 2.1 se puede observar el planteamiento de los clasificadores basados en aprendizaje automático, utilizando un proceso general de inducción durante el aprendizaje y de deducción para la clasificación de nuevos documentos. Más adelante, mostraremos además resultados respecto a la efectividad de estos clasificadores, que se asemeja mucho a la efectividad alcanzada por humanos.

Ejemplos de este tipo de clasificadores son *Hermes* y *Poesía*. El primero es un emisor de noticias y el segundo un supervisor de páginas web. Se puede obtener información detallada sobre estos sistemas en [Giraldez et al., 2002] y [Gómez et al., 2002] respectivamente.

En la literatura se pueden encontrar numerosas publicaciones sobre aprendizaje semisupervisado [Chapelle et al., 2006; Nigam et al., 2006] y no supervisado [Ko y Seo, 2000;

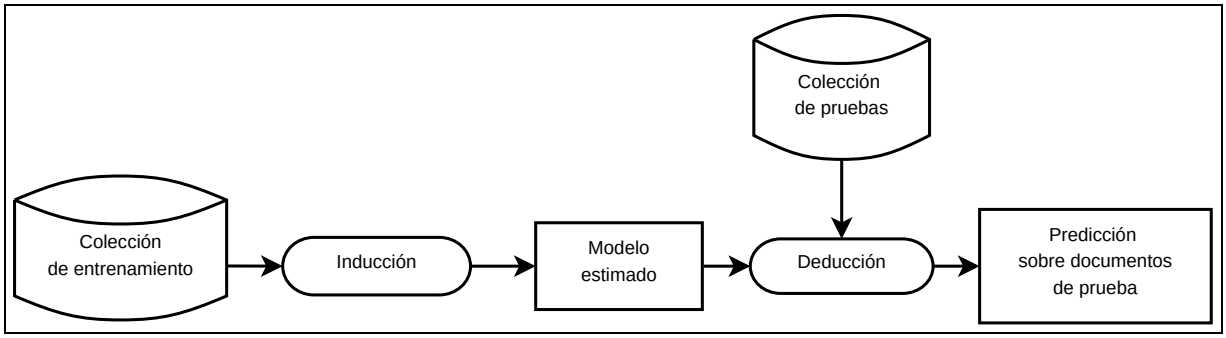


Figura 2.1: Aprendizaje inductivo [Mendialdua, 2008].

El-Yaniv y Souroujon, 2001; Adami et al., 2003], donde se utilizan ejemplos no etiquetados para el entrenamiento. Sin embargo, dejaremos dicho concepto fuera de esta tesis, y nos centraremos en los métodos supervisados, pues para realizar todos nuestros experimentos sólo emplearemos corpus cuyos documentos y categorías se encuentran determinados de antemano.

2.3. Construcción de un clasificador inductivo-deductivo

Como ya se ha comentado, hablar de la construcción de un clasificador mediante aprendizaje automático implica a su vez hablar de aprendizaje supervisado, pues se aprende de un conjunto de textos ya clasificados previamente, utilizando la información de pertenencia de cada documento a ciertas categorías. Partiendo de estas muestras como ejemplos de las futuras entradas, se puede desarrollar un clasificador principalmente en dos fases:

1. Preproceso de documentos. Puesto que los documentos de texto no suelen ser adecuados para utilizarlos directamente como entrada del algoritmo de aprendizaje, resulta necesario transformarlos mediante algún tipo de preprocesamiento. Éste consiste en la creación de representaciones internas de los documentos, un proceso que se realiza a su vez en dos pasos:

   - Obtención de características. De entre toda la información que nos ofrecen los documentos de entrenamiento, se identifican aquellas características que pueden resultar más relevantes para la tarea de clasificación. Como resultado de este proceso podemos obtener por ejemplo elementos tales como palabras o frases, así como métricas de la longitud de las mismas, o también indicadores de su posición, o de la frecuencia de aparición dentro del documento o en la colección de entrenamiento.
   - Ponderación de características. Se calculan distintas valoraciones o pesos para cada una de las características, basándose principalmente en la contribución a la hora de discriminar entre categorías que aporta cada una.

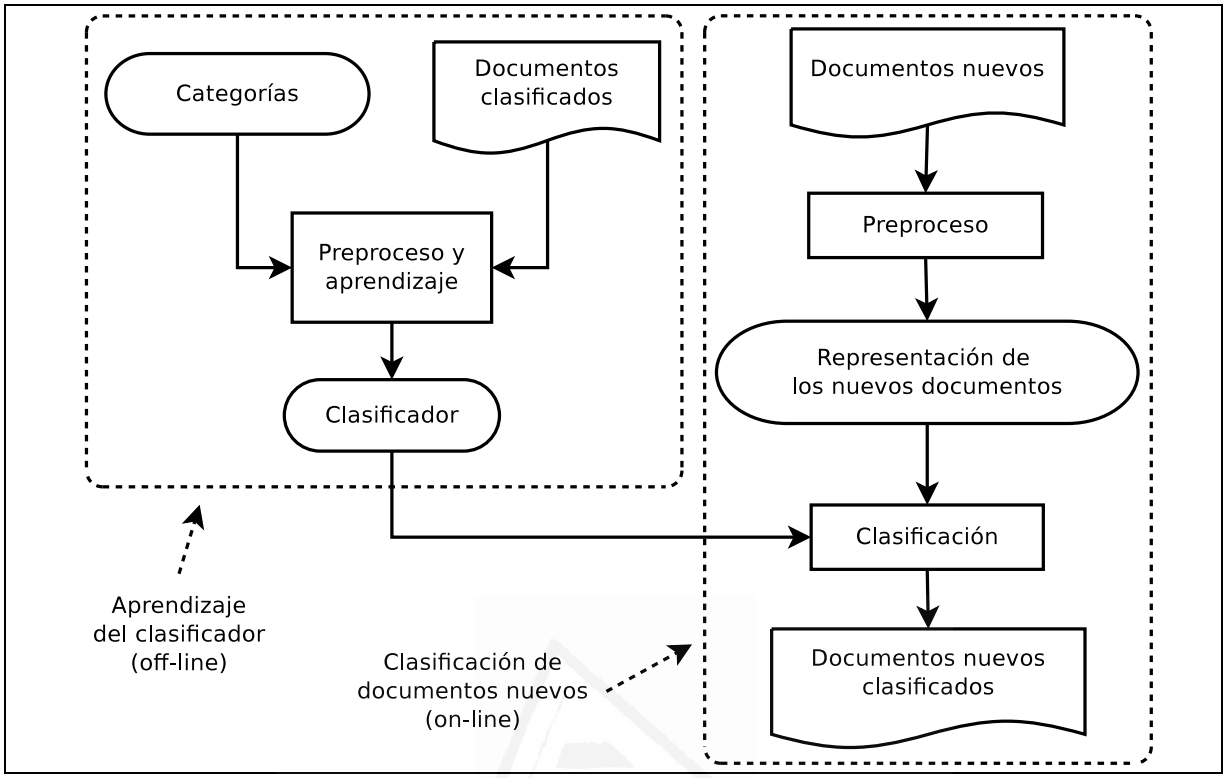


Figura 2.2: Representación gráfica del proceso de aprendizaje supervisado y clasificación automática de nuevos documentos [Valero, 2005].

2. Aprendizaje del clasificador. Se desarrolla un clasificador partiendo de las representaciones internas creadas a partir de las características extraídas de los documentos de entrenamiento en la fase anterior.

Universitat d'Alacant
Universidad de Alicante

La figura 2.2 muestra gráficamente estos pasos, indicando cómo se pueden aprovechar documentos ya categorizados para entrenar al clasificador y, cómo después de una etapa de aprendizaje que se realiza de manera previa a la clasificación (lo que se conoce como proceso *off-line*), se pueden introducir y procesar documentos inéditos que serán evaluados emulando la asignación de clases realizada previamente por expertos en el área de aplicación.

A continuación vamos a detallar cada uno de los elementos que intervienen en la construcción del clasificador: taxonomías, corpus de trabajo, características de aprendizaje y algoritmos de clasificación.

2.3.1. Taxonomías

Una taxonomía es un tipo de vocabulario controlado en el que los términos identifican clases o categorías que agrupan elementos que tienen ciertas propiedades en común. Dichos términos pueden encontrarse organizados en un mismo nivel o pueden estar conectados

mediante algún modelo estructural (por ejemplo, utilizando jerarquías). El objetivo de todo clasificador es asignar a los elementos de entrada alguna de las clases de una taxonomía concreta definida en el dominio de aplicación del sistema.

Para obtener una taxonomía, se deben tener en cuenta las siguientes consideraciones [Centelles, 2005]:

- El área de conocimiento a representar se debe encontrar bien delimitado.
- Se deben especificar cuáles son las fuentes que establecerán el conjunto de términos: proporcionados por especialistas, extraídos de documentos representativos, o de nomenclaturas existentes.
- Se deben identificar los diferentes términos que designan un mismo concepto.
- Se debe establecer un esquema de organización de los términos o categorías, especificando qué criterios se siguen para agrupar o dividir las categorías.

En muchas ocasiones el vocabulario se encuentra especialmente orientado a los sistemas de navegación, organización y búsqueda de contenidos en Internet. Podemos mencionar por ejemplo el *Open Directory Project*, el directorio jerárquico de páginas web más extenso y completo que existe, mantenido de forma manual y accesible libremente.[4]

2.3.2. Corpus de trabajo

La forma en que un clasificador automático aprovecha las decisiones previas tomadas por expertos humanos puede diferir en gran medida de un clasificador a otro. Sin embargo, la mayoría utilizan un proceso general inductivo que establece una serie de formas de actuar que repetidas sobre documentos nuevos pueden llevarles a clasificarlos correctamente. Dicho de otra forma, existe una fase previa a la clasificación durante la cual el clasificador lleva a cabo un proceso de aprendizaje del que se infieren una serie de condiciones de pertenencia que un documento cualquiera debería satisfacer para poder ser considerado como clasificable en una categoría determinada [Jiménez, 2007].

Clasificar se reduce entonces a evaluar ciertas condiciones de pertenencia para un documento y una clase en concreto, emitiendo una decisión afirmativa o negativa en cada caso. Durante este proceso, el clasificador desconocerá por completo el significado de los documentos o las clases a las que se pueden asignar. Sin embargo, debemos confiar en que la representación de los documentos a partir de la selección y ponderación de las características que los componen puede llegar a contener de forma implícita la esencia de las materias que traten. Si esto es así, una vez extraído el contenido, a través de comparaciones se podrá

[4] Accesible desde http://www.dmoz.org/. A finales del 2012 ya cuenta con más de cinco millones de páginas web organizadas en más de un millón de categorías.

establecer la cercanía temática que indique la pertenencia de un nuevo documento a una o varias clases, ya que el entrenamiento del clasificador le permitirá apreciar determinadas características para asociarlas con una o varias categorías.

La mayor parte de los clasificadores llevan a cabo algún tipo de entrenamiento previo antes de enfrentarse a tareas reales de categorización. Para llevar a cabo la fase de entrenamiento es necesario disponer de conocimiento previo de expertos en forma de decisiones de categorización asignadas a documentos. Esto se materializa en un conjunto de documentos preclasificados y etiquetados para que el sistema pueda leer las clases a las que pertenecen cada uno de ellos.

Este corpus de documentos preclasificados se denomina *colección de entrenamiento* y, para poder evaluar la fase de aprendizaje, se suele dividir antes de ser utilizado. Lo más normal es realizar una división en dos subconjuntos: *entrenamiento* y *prueba*, donde el conjunto de entrenamiento sirve para educar al clasificador, y el de prueba para medir la efectividad conseguida, teniendo en cuenta que dichos conjuntos deben ser disjuntos.

Además, también es muy común reservar ciertos documentos para realizar un ajuste de los parámetros del algoritmo antes de llevar a cabo la evaluación final (*corpus de validación*). De esta forma, se consideran finalmente tres conjuntos de documentos: uno de aprendizaje, otro de validación y uno adicional reservado para evaluar finalmente la calidad de los resultados. Esta división da lugar a tres fases principales en el desarrollo de un clasificador: aprendizaje, validación y evaluación. Las describiremos a continuación.

Aprendizaje

El aprendizaje se realiza mediante un proceso general de inducción, cuyo objetivo es conseguir un modelo en forma de condiciones de pertenencia, que luego se utilizará en una fase posterior de clasificación. Ambos procesos llevan perfeccionándose durante muchos años, pues los primeros clasificadores que fueron capaces de efectuar decisiones correctas sobre documentos nuevos en base a conocimiento previo datan de los años 1960.[5] Sin embargo, como ya comentamos anteriormente, en las primeras soluciones la codificación del conocimiento en forma de condiciones de inclusión se realizaba de forma completamente manual, lo que incrementaba mucho los costes debido a la cualificación necesaria y a la cantidad de horas de trabajo requeridas.

Por ello, en la actualidad, la mayoría de los esfuerzos se vuelcan en el proceso de aprendizaje, para generar de forma automática las condiciones de pertenencia a las clases, analizando los documentos preclasificados y las decisiones de los expertos humanos. Por otro lado, han adquirido menos importancia las decisiones de clasificación, pues son mucho más fáciles de tomar y frecuentemente se llevan a cabo mediante la aplicación de funciones de relevancia que se basan en medidas de similitud.

[5] Nos referimos a los clasificadores basados en ingeniería del conocimiento, que introdujimos en la sección 2.2.1.

También es importante hacer referencia al tamaño del corpus, que debe ser lo suficientemente representativo de las situaciones con las que posteriormente se puede encontrar el clasificador. En [Massey et al., 2003] se afirma que los algoritmos basados en aprendizaje automático tienden a alcanzar la máxima precisión a medida que se incrementa el tamaño del corpus de entrenamiento. Por ejemplo, en el mismo trabajo se muestra la influencia del tamaño del corpus dentro del ámbito del filtrado de correo spam, observando la importancia de disponer de suficientes ejemplos para aprender qué correos pueden ser clasificados como spam de manera fiable.

Validación

Utilizando el modelo obtenido durante la fase de aprendizaje, la clasificación de un documento nuevo consiste en la evaluación de las condiciones establecidas para cada categoría, considerando la asignación o exclusión dependiendo de si cumple con las condiciones de pertenencia. De esta forma el proceso de clasificación se lleva a cabo por lo general de forma muy rápida, pues tal como se ha comentado, la mayor parte del peso recae sobre el proceso de aprendizaje, después del cual, ya podremos obtener algunos resultados, aunque probablemente, en una primera aproximación no sean los esperados.

Con el objeto de mejorar las deficiencias que podamos apreciar, debemos llevar a cabo un proceso de validación, que básicamente consiste en evaluar el comportamiento del clasificador ya "enseñado" sobre un conjunto de documentos que no intervinieron durante la fase de aprendizaje. El hecho de que todos los documentos estén clasificados de antemano permite comprobar el nivel de correspondencia entre las decisiones del clasificador y las decisiones óptimas,[6] indicándonos si debemos ajustar los parámetros del clasificador. Se puede variar por ejemplo el modelo de representación, o el umbral a partir de cual se considera que un documento forma parte de una categoría. Finalmente, después de aplicar ciertos ajustes, las pruebas se pueden repetir para observar si los resultados obtenidos son más correctos que los anteriores.

Evaluación

Una vez se han llevado a cabo las fases de aprendizaje y validación se mide de nuevo la efectividad del clasificador. Para esto se suelen emplear medidas de evaluación estándar como la precisión y la cobertura[7] sobre un conjunto similar a los anteriores cuyos documentos sin embargo no se han utilizado hasta ahora. Al obtener una medida real de la efectividad del clasificador, podremos decidir si los resultados tienen la suficiente calidad para ser utilizados en aplicaciones reales.

La técnica de validación cruzada (*k-fold cross validation*) también puede emplearse para

[6] Se supone que los expertos humanos siempre toman las decisiones correctas, aunque en la realidad esto no siempre es así.

[7] Estos conceptos se definirán más adelante, en la sección 5.2.1.

evaluaciones cuando el conjunto de documentos etiquetados es pequeño. Esta técnica consiste en dividir el conjunto de entrenamiento en k conjuntos, generando k versiones del clasificador, cada una entrenada con un conjunto de datos diferente, utilizando los documentos de entrenamiento de los otros clasificadores como conjunto de datos de prueba. Más concretamente, se utilizarán como entrenamiento los textos contenidos en $k-1$ conjuntos y sólo se empleará un conjunto para evaluación. Además, la precisión conseguida no tiene por qué verse afectada, pues el resultado final se obtiene promediando los resultados individuales. Sin embargo, esta consideración no suele aplicarse en clasificación de textos si el corpus inicial es lo suficientemente grande [Sebastiani, 2002].

2.3.3. Características de aprendizaje

Una característica puede ser cualquier elemento o propiedad que pueda ser considerada como un atributo representativo de los documentos del corpus. Dada la ambigüedad de este concepto, es de suma importancia determinar cuáles son las características más relevantes, es decir, aquellas que realmente puedan contribuir a identificar la pertenencia de los documentos a las categorías.

En [Scott, 1998] se plantea un ejemplo muy genérico que nos ayudará a entender la importancia de una buena elección de las características. El objetivo propuesto consiste en identificar aquellos rasgos que nos permitan clasificar automáticamente personas en dos conjuntos: bailarines de ballet y jugadores de rugby. Debemos decidir cómo representar a los individuos para que un sistema pueda distinguirlos de la mejor forma posible. Por ejemplo, si elegimos observar el color del pelo y el color de los ojos, llegaremos a la conclusión de que no siguen norma alguna para ninguno de los dos tipos. También se propone identificar el índice de grasa corporal y el tamaño de los zapatos. En este último caso, los individuos de ambos conjuntos tienen un índice muy bajo de grasa corporal, pero sin embargo, el tamaño de sus zapatos sí difiere de un conjunto a otro, y por lo tanto se puede considerar una peculiaridad válida. Por último, se propone tener en cuenta además la altura y el peso, que sí parecen ser características bastante identificativas, pues los jugadores de rugby tienen una constitución física muy diferente de los bailarines de ballet.

Si observamos simples documentos de texto y tratamos de extraer las peculiaridades más representativas de cada uno, nos encontraremos con que la situación no es tan trivial como en el ejemplo anterior. Es más fácil encontrar aquellas características que nos permiten distinguir personas u objetos que encontrar semejanzas o diferencias entre textos escritos en lenguaje natural. La mayoría de los clasificadores actuales extraen las palabras que se encuentran en los documentos y consideran cada palabra como una característica individual. Este método es conocido como *bolsa de palabras*, donde una característica será cualquier cadena de caracteres alfanuméricos delimitados por espacios en blanco o signos de puntuación. Esto simplifica mucho el proceso de representación, pues de esta forma, utilizando sólo unos y ceros podemos llegar a representar cada documento como una sucesión de valores binarios, indicando la presencia (unos) o ausencia (ceros) de todas las posibles palabras. Por ejemplo,

si nos encontramos con tres documentos que contienen el siguiente texto:

- Documento 1: Enfermedades de las válvulas cardíacas.
- Documento 2: Prótesis de válvulas cardíacas.
- Documento 3: Sistema de conducción cardíaco.

la representación de bolsa de palabras básica sería la siguiente:

Palabras	Documento 1	Documento 2	Documento 3
enfermedades	1	0	0
de	1	1	1
las	1	0	0
válvulas	1	1	0
cardíacas	1	1	0
prótesis	0	1	0
sistema	0	0	1
conducción	0	0	1
cardíaco	0	0	1

Este planteamiento posiblemente resultaría válido si los documentos que tratasen de temas diferentes estuvieran formados por palabras diferentes, y los documentos que tratasen del mismo tema tuvieran muchas palabras en común. Sin embargo, esto no es así en la realidad y, además, la complejidad del lenguaje humano puede ocasionar que esta representación tan simple no sea la más adecuada. Concretamente, nos podemos encontrar con las siguientes dificultades:

- Una misma palabra puede representar diferentes conceptos (polisemia).
- Un mismo concepto puede venir representado por palabras diferentes (sinonimia).
- Los conceptos pueden venir representados por parejas de palabras o incluso combinaciones de varias palabras.
- Una palabra puede resultar más o menos relevante dependiendo de cuántas veces aparezca, o a qué parte de la oración represente.
- Algunas palabras pueden no aportar significado alguno (*palabras de parada* o *stopwords*).

- La cantidad de características a considerar puede resultar demasiado grande para tratar de emplearlas todas en la resolución de problemas reales.[8]

Algunos de estos problemas surgen porque mediante la representación de bolsa de palabras se pierde mucha información del texto original. Por ejemplo, el orden de las palabras no se tiene en cuenta, y la estructura de las oraciones y los párrafos desaparece. Es decir, el texto deja de ser legible para nosotros por intentar hacerlo legible para el algoritmo de aprendizaje. Por ello algunos autores proponen soluciones intermedias y sugieren utilizar frases como características [Scott y Matwin, 1999]. Por ejemplo, la expresión "Sistema Nervioso Simpático" está formada por tres palabras que utilizadas de forma individual tienen un significado tan genérico que probablemente carezcan de validez para la clasificación. Manteniéndolas juntas como una única característica conseguiremos capturar mayor cantidad de información, pero también ampliaremos el tamaño del conjunto de características y, por lo tanto, se agudizará la necesidad de utilizar ciertas técnicas de selección y filtrado.

En cualquier caso, sea cual sea la representación a utilizar, es necesario preprocesar los documentos para conseguir una representación compacta del conjunto de documentos preclasificados. Puesto que el contenido de esta tesis se basa en el procesamiento de textos, a partir de ahora hablaremos de *términos* o *características* indistintamente, para referirnos a palabras o grupos de palabras extraídos de los documentos. Si consideramos el caso más sencillo utilizado en la literatura, en el que las características sean simples palabras, el conjunto de todas las características se denomina *vocabulario*. Nosotros además utilizaremos este concepto para designar el conjunto de todos los términos, sean palabras o agrupaciones de varias palabras.

También nos podemos encontrar con ciertos atributos o propiedades de los términos, los documentos o las categorías, que pueden formar parte de las características: el autor del documento, la longitud o posición de un término, el número de apariciones, la coincidencia de un término con el descriptor de una categoría, etc. En esta tesis tales datos no serán considerados como características en sí, sino que los utilizaremos para valorar los términos y dar en consecuencia mayor o menor relevancia a cada uno de ellos.

Cuando hayamos decidido qué información debe ser considerada de cada documento, éste ya se puede transformar en un vector de características de entrada para el algoritmo de clasificación. Más formalmente, dado un documento de texto d_i, su representación vectorial viene dada por $d_i = \{w_1, w_2, \cdots, w_n\}$, donde cada dimensión corresponde a una característica diferente en la colección de documentos, siendo w_j el peso asignado a la característica j. La construcción del vector de características se puede llevar a cabo mediante una serie de pasos [Lewis, 1992c]:

1. Normalización del texto.

[8] Es importante clarificar que en un primer paso debemos determinar qué partes o peculiaridades del texto son características potenciales, no debiendo decidir a priori si dichas características serán finalmente utilizadas o no. Una vez obtenido el conjunto que las incluya todas, se puede realizar un filtrado para eliminar las que se consideren menos relevantes.

2. Obtención de características.

3. Generación del vector de características.

4. Reducción de la dimensionalidad.

El paso de *normalización del texto* transforma cualquier tipo de documento en una secuencia de tokens de los que se puede extraer un conjunto de características. Si consideramos el caso más sencillo, podemos optar por agrupar todas las palabras distintas que encontremos en los documentos de entrenamiento para generar un conjunto de términos candidatos que puedan ser usados como vocabulario. Este proceso se conoce como *obtención de características*. En el paso de *generación del vector* se calculan y ajustan los pesos para cada una de las características de cualquier documento. Y finalmente, puesto que el conjunto de características suele ser muy grande, el siguiente paso será reducir el tamaño de dicho conjunto en un proceso conocido como *reducción de la dimensionalidad*.

En la figura 2.3 se puede observar la secuenciación de dichos procesos aplicados tanto para los documentos de aprendizaje como para nuevos documentos todavía sin clasificar. Se muestran ejemplos sencillos de documentos que se transforman en vectores de características considerando como tales las palabras que allí aparecen. También se muestra la frecuencia de aparición de cada palabra, que puede ser utilizada por ejemplo para ponderar cada elemento del vector según el número de apariciones. Este es, sin embargo, uno de los casos más sencillos, pues en la realidad existen muchas aproximaciones que, por ejemplo, agrupan varias palabras para formar nuevas características. También se puede considerar información adicional como la longitud de las mismas o la posición en la que aparecen dentro del texto (título, resumen, etc.).

Universitat d'Alacant
Universidad de Alicante

Es conveniente matizar que la reducción de la dimensionalidad suele realizarse una vez se han construido los vectores de características, pues en ese momento ya se dispone de toda la información que condensará la representación final de los documentos y, por lo tanto, se puede evaluar más objetivamente la técnica o el umbral de filtrado más adecuado. Por ejemplo, a continuación enumeramos algunas de las consideraciones que se suelen utilizar habitualmente:

- Es común eliminar los sufijos de las palabras, para reducir a una característica común varias palabras que tengan el mismo significado (*stemming*). Por ejemplo, las palabras "anestesia", "anestesiar", "anestesiología", "anestésicos", etc, podrían ser reducidas a su raíz común "anestesia", consiguiendo en consecuencia reducir la cantidad total de características a utilizar.

- Se suelen identificar y eliminar aquellas palabras carentes de significado (*palabras de parada* o *stopwords*), tales como artículos, preposiciones, etc.

- También aquellas palabras que aparecen con mucha frecuencia en muchos documentos pueden resultar prescindibles, pues representan características muy genéricas y, por lo tanto, de poco valor discriminativo entre clases.

Para poder entender el papel de cada una de las cuatro etapas, en las próximas secciones comentaremos con más detalle cada una de éstas.

Normalización del texto

El objetivo de la normalización del texto es convertir los ficheros de entrada en una secuencia de elementos lingüísticos conocidos como tokens. La idea es aislar y unificar aquellos elementos que contengan información, pues en el siguiente paso, los tokens generados se utilizarán para obtener características. Debido a la variedad de formatos y las diferencias de escritura que se suelen encontrar en los textos de entrada, dicho proceso se realiza habitualmente en dos pasos: uno de limpieza de los documentos y otro de unificación de criterios para tratar de la misma forma tokens que identifican el mismo término a pesar de estar escritos de forma diferente.

Primero debemos reconocer los diferentes componentes textuales para cada uno de los formatos de los ficheros, teniendo en cuenta que dichos ficheros pueden ser de muy diversos tipos (correos electrónicos, ficheros LATEX, postscript, documentos HTML, etc.), y ello implica cierto procesamiento. Por ejemplo, si se procesan correos electrónicos, habitualmente se extraen el campo del asunto y el cuerpo del correo, o para documentos HTML, primero se deben eliminar todas las etiquetas. Una vez seleccionados los campos a utilizar o eliminadas ciertas palabras clave o sentencias de formato o control, debemos considerar como válidas aquellas cadenas de caracteres que se encuentren delimitadas por ciertos elementos, tales como espacios en blanco, tabuladores o saltos de línea.

En segundo lugar, la secuencia resultante de tokens se debe normalizar dependiendo de la aplicación. Habitualmente, todas las letras se convierten a minúsculas y se eliminan todos los signos de puntuación. Además, los tokens que contienen cualquier carácter no alfanumérico suelen ser eliminados. A menudo tampoco se consideran los tokens que contienen valores numéricos. Alternativamente, todos los dígitos podrían ser mapeados en un dígito predeterminado (por ejemplo, "1"). Siguiendo otro enfoque de la normalización del texto, entidades tales como nombres propios de personas, localizaciones, organizaciones o nombres de productos, podrían ser identificadas y marcadas como tokens individuales. Por ejemplo, el fragmento de texto "Vitamina B12" podría ser considerado como un solo elemento y ser identificado como cierto tipo de medicamento. En este caso, los dígitos contenidos por la expresión no serían normalizados o eliminados. Hay que tener en cuenta que el añadir esta peculiaridad al proceso de normalización requeriría cierto dominio y conocimiento del ámbito de aplicación del clasificador y, por lo tanto, no consideramos oportuno tratarlo en esta tesis.

Obtención de características

Una vez tengamos las secuencias de tokens normalizados extraídos de cada documento, se debe llevar a cabo la tarea de extraer los términos para producir una secuencia de características basada en estos tokens, dependiendo de si ya hemos procesado todos los documentos

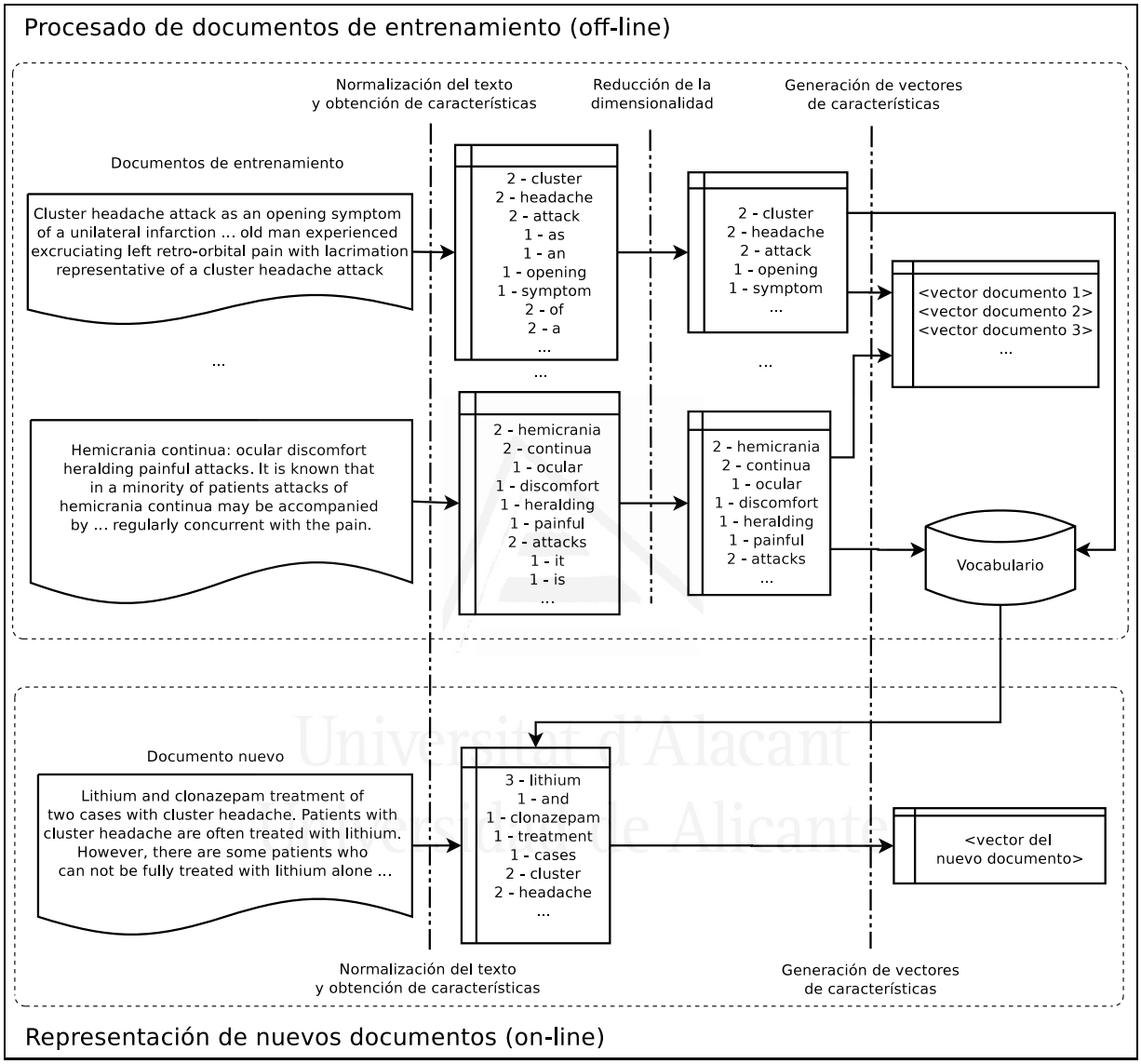


Figura 2.3: Representación de documentos. Debido a las limitaciones de espacio, se ha omitido intencionadamente parte del texto de los documentos de entrenamiento originales. Además, la reducción en la dimensionalidad se limita a eliminar del texto simplemente las palabras de parada. Aún así, este ejemplo puede llegar a ser representativo, pues nos permite observar el proceso de elaboración de los vectores de documentos, utilizando palabras como características y prescindiendo de aquellas que no aportan mayor calidad a la abstracción de los documentos.

de entrenamiento y hemos obtenido por lo tanto el conjunto delimitado del vocabulario.

Por un lado, si ya disponemos de un vocabulario, se puede procesar fácilmente cualquier documento omitiendo todas las características que no se encuentren en el vocabulario. Por ejemplo, si procesamos el documento d_i y consideramos como peso de cada término su frecuencia de aparición (*term frequency* - tf), se obtendría la siguiente representación: $d_i = \{tf_1, tf_2, \ldots, tf_n\}$, donde n representa el número de términos en el vocabulario y tf_j representa el número de veces que el término j aparece en el documento d_i. Dado este vector, ya se podría aplicar alguna técnica sencilla de reducción de dimensionalidad para proceder al paso de construcción de vectores.

Por otro lado, si no disponemos de un vocabulario predeterminado, las salidas del proceso de obtención de características para cada documento de entrenamiento se deben unir para generar un conjunto de términos único que pueda constituir el conjunto del vocabulario. El tamaño del mismo puede variar mucho, dependiendo de la definición de los términos y del tamaño y temática de la colección de documentos, pero en muchos casos es bastante común disponer del orden de cientos de miles de características. Puesto que la aplicación de los algoritmos de aprendizaje automático se limita mucho cuando el número de características es muy grande, el próximo objetivo debe ser obligatoriamente reducir el tamaño final del conjunto del vocabulario en una fase de reducción de dimensionalidad.

Pero antes de abordar dicha fase, podemos prevenir la generación de un conjunto excesivamente grande del vocabulario eligiendo correctamente los términos a utilizar. Para ello debemos plantearnos qué elementos lingüísticos deberíamos considerar como características. Un reto importante en el procesamiento y comprensión del lenguaje natural es que la información o el significado transmitido siempre depende del contexto. Por norma, la mayoría de los textos no se deberían interpretar literalmente pues, por ejemplo, una palabra puede tener diferentes significados o una frase puede expresar diferentes ideas dependiendo del contexto en el que aparezca. Esto anticipa que el intentar representar documentos escritos en lenguaje natural por medio de un conjunto de términos puede ser una tarea difícil. Existen gran variedad de enfoques para intentar capturar (o ignorar) hasta cierto punto el significado dependiendo del contexto, pero en este trabajo nos centraremos en los cuatro más comunes:

- Nivel grafémico: análisis partiendo de letras.
- Nivel léxico: análisis partiendo de palabras individuales.
- Nivel sintáctico: análisis partiendo de la estructura de las frases.
- Nivel semántico: análisis relacionado con el significado de las palabras o frases.

En los dos primeros niveles sólo se trabaja con elementos estadísticos extraídos del texto, como por ejemplo, frecuencias de aparición de letras o palabras. Pero las representaciones basadas en dichas frecuencias no pueden capturar completamente el significado de los documentos. De hecho, sólo existe una débil relación entre ocurrencias de términos y el contenido

del documento. Se asume que las frecuencias de términos de un documento simplemente sugieren su contenido. Niveles más altos de análisis del texto intentan capturar más contenido semántico explotando e incrementando cierta cantidad de información contextual tal como la estructura de oraciones, párrafos o documentos. En general, las técnicas de procesamiento de lenguaje natural pueden llegar a proporcionar una representación más rica mediante un análisis sintáctico y semántico de los documentos.

Generación del vector de características

Una vez obtenidas las estadísticas de aparición de todos los términos para todos los documentos de aprendizaje, la tarea de generar el vector consiste en el paso de crear un vector de pesos $d = \{w(t_1), w(t_2), \ldots, w(t_n)\}$ para cualquier documento d, donde n es el número de elementos en el vocabulario. Cada peso $w(t_i)$ expresa la importancia del término t_i en el documento d con respecto a sus frecuencias de aparición en todos los documentos de entrenamiento. El objetivo de utilizar un peso en vez de utilizar las frecuencias directamente es mejorar la efectividad de la clasificación [Nigam y Ghani, 2000]. La ponderación se calcula habitualmente mediante una función que combina la frecuencia de aparición de un término particular en un determinado documento y la frecuencia de aparición en toda la colección de documentos. Variando esta función, podemos producir diferentes pesos de términos que pueden desembocar en diferentes modelos, tal como se verá en la próxima sección.

En [Salton et al., 1988] se identifican tres componentes principales para determinar la importancia de un término: local, global y normalización[9]. Basándose en estos tres factores, se puede calcular el peso de un término t para un documento d de la siguiente forma:

$$w(t,d) = \frac{w_{local}(t,d) \cdot w_{global}(t)}{w_{norm}(d)} \quad . \tag{2.1}$$

Más adelante, en la sección 4.2.5, analizaremos algunas de las propuestas existentes para obtener cada uno de los componentes. A continuación introduciremos los conceptos básicos y sus fundamentos teóricos:

- **Componente local**. El factor de ponderación local $w_{local}(t,d)$ refleja la importancia del término t para un documento d en particular. Recordemos que, por ejemplo, la frecuencia de un término se puede considerar como un indicador válido de la importancia del mismo. La investigación en recuperación de información sugiere que los términos que se utilizan frecuentemente y que, por lo tanto, tienen una alta frecuencia de aparición, deben ser enfatizados para mejorar la efectividad [Salton et al., 1988]. Típicamente, el factor de peso local se obtiene aplicando una función de transformación a la frecuencia del término en caso de que el término t aparezca en el documento d:

[9] Se puede consultar [Dumais, 1991] para ampliar los conceptos de peso local y global.

$$w_{local}(t,d) = \begin{cases} f(TF(t,d)) & \text{si } TF(t,d) > 0 \\ 0 & \text{en otro caso} \end{cases} , \tag{2.2}$$

donde $TF(t,d)$ representa el número de veces (frecuencia) que aparece el término t en el documento d. Se puede observar que si el término t no aparece en el documento d, el peso $w_{local}(t,d)$ es 0 y, por lo tanto, el peso final también lo será independientemente del factor de ponderación global.

Dos funciones de transformación inmediatas son, por ejemplo, la función identidad, que no cambia las frecuencias de término proporcionadas y el indicador binario, que devuelve 1 para una frecuencia de término distinta de cero y 0 en otro caso. Funciones más elaboradas podrían normalizar las frecuencias de los términos mediante una división por la frecuencia de término máxima dentro de cada documento. Otra posibilidad es aplicar un logaritmo[10] a la frecuencia de término, de tal forma que una ocurrencia adicional del término t en el documento d sea considerada más importante a niveles de frecuencias de término pequeñas que a niveles más grandes [Solomon y Howes, 1951].

- **Componente global**. El factor de ponderación global $w_{global}(t)$ toma en consideración la importancia del término t dentro de todo el conjunto de los documentos de entrenamiento. Como se ha comentado anteriormente, un término se considera más importante si aparece en unos pocos documentos. Dar un peso más alto a unos términos muy específicos nos permite rechazar documentos no relevantes y por lo tanto, nos permite enfatizar la precisión [Salton et al., 1988]. Por ejemplo, un factor de ponderación global ampliamente utilizado es la *inversa de la frecuencia del documento* definida como $log\frac{n}{n(t)}$, donde n representa el número total de documentos y $n(t)$ el número de documentos en los que el término t aparece.

- **Componente de normalización**. El denominador de la ecuación 2.1 $w_{norm}(d)$ es un factor de normalización. Al normalizar los pesos del vector que representa un documento d se pueden abstraer dichos pesos de las diferencias en las longitudes de los documentos. Típicamente se aplica una normalización a la longitud unitaria, también conocida como normalización del coseno [Singhal et al., 1996]. Sin embargo, en un contexto probabilístico, es más habitual normalizar por la suma de todos los componentes del vector de pesos de cada documento [Salton et al., 1988].

  Los documentos más grandes tienden a contener más características diferentes. Con un mayor número de características presentes en un documento, la posibilidad de encontrar términos que aparezcan en otros documentos es mayor. Consecuentemente, los documentos más grandes tienen una mayor probabilidad de ser similares a otros documentos si no están normalizados [Salton et al., 1988]. Además, en muchos casos, para optimizar los cálculos de similitud, se normalizan los pesos antes de abordar ningún tipo de aprendizaje, pues resulta computacionalmente más eficiente normalizar todos

[10] El uso del logaritmo para el cálculo de pesos será justificado más adelante, en la sección 3.3.1.

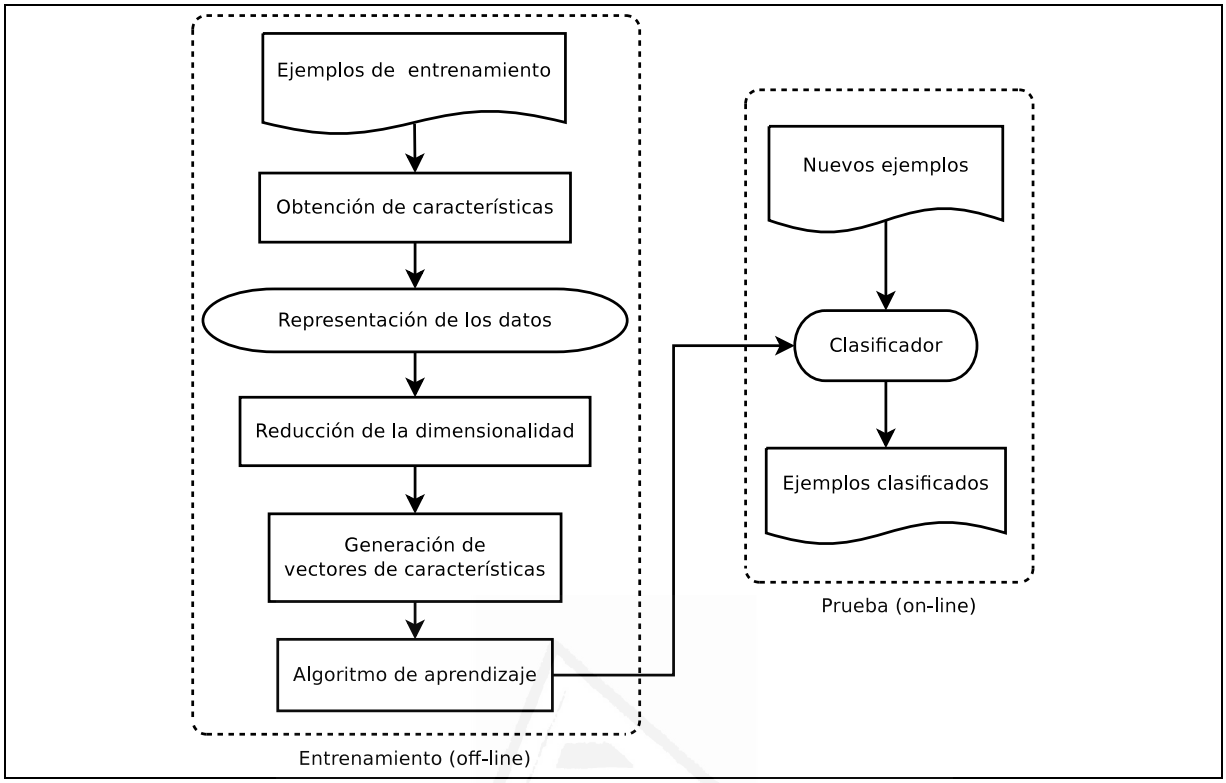


Figura 2.4: Selección de términos para reducir la dimensionalidad antes del proceso de aprendizaje.

los vectores de pesos, en vez de realizar normalizaciones cada vez que un documento esté implicado en el cálculo de una función de similitud.

Universitat d'Alacant
Universidad de Alicante

Reducción de la dimensionalidad

La reducción de la dimensionalidad consiste en disminuir el tamaño del vector que contiene las características representativas de un documento. Este proceso se suele realizar off-line durante el proceso de aprendizaje, tal como se muestra en la figura 2.4.

Es posible que se disponga de un vocabulario establecido de antemano, que puede haber sido acotado por expertos humanos. En este caso, la reducción de dimensionalidad no es necesaria, pues ya se dispone de un conjunto restringido de términos que limita las características a extraer de los documentos para obtener una representación adecuada que minimice los recursos computacionales requeridos durante el proceso de aprendizaje. Sin embargo, si la lista de términos se obtiene automáticamente procesando los documentos de entrenamiento, habrá que tener en cuenta que, de esta forma, el conjunto del vocabulario que se suele obtener es habitualmente muy grande. En este último caso el objetivo inmediato será rebajar el número de características que se usen al final para representar los documentos, bajo la premisa de que el conjunto resultante de características sea todavía discriminativo para las diferentes categorías. Controlar la dimensionalidad es esencial por dos motivos:

- La complejidad de numerosos algoritmos depende crucialmente no sólo de los ejemplos de entrenamiento, sino también del número de características, hasta el punto de que en algunos casos no sea sólo conveniente, sino necesario, reducir el número de términos [Dash y Liu, 1997].

- Aunque disponer de más características puede entenderse como sinónimo de mayor cantidad de información representada y, por lo tanto, debería desembocar en una mayor precisión en la clasificación, esto no siempre es así en la práctica. Un número mayor de características puede acarrear que muchas de ellas sean irrelevantes y entorpezcan el aprendizaje del algoritmo, generando un problema conocido como "maldición de la dimensionalidad" [Duda y Hart, 1973]. Dado un número fijo de ejemplos de entrenamiento, los resultados teóricos y empíricos en el aprendizaje automático demuestran que a menudo existe un número máximo de características a partir del cual la efectividad de un clasificador inducido empieza a descender [Lewis, 1992b]. Por lo tanto, eliminar las características menos informativas puede en realidad incrementar el rendimiento del clasificador.

2.3.4. Algoritmos de clasificación

En la literatura podemos encontrar gran variedad de algoritmos de aprendizaje y clasificación, pues se han dirigido numerosos esfuerzos para recopilarlos, estudiarlos y compararlos [Dumais et al., 1998; Yang y Liu, 1999; Sebastiani, 2002]. A continuación describiremos varias formas de abordar el aprendizaje supervisado, todas ellas ampliamente utilizadas y diferenciadas, pues aunque se basan en paradigmas completamente diferentes conducen a resultados similares. Primero introduciremos la base teórica de aquellos algoritmos que funcionan por similitud o distancia, tales como el k-vecino más cercano, muy difundidos en la actualidad y a los que nos referiremos como clasificadores basados en ejemplos. En segundo lugar, hablaremos de las máquinas de vectores de soporte, un tipo de separador lineal que a pesar de su relativa complejidad ofrece una alta precisión. En tercer lugar expondremos la idea de los clasificadores probabilísticos, tales como Naive Bayes, muy comunes para clasificación de textos. Por último hablaremos de los clasificadores basados en redes neuronales y árboles de decisión, que se diferencian considerablemente de los anteriores, ya que no persiguen la resolución de un modelo matemático concreto.

Clasificadores basados en ejemplos

Este tipo de clasificadores representan cada clase mediante uno o varios prototipos, utilizando el mismo lenguaje representacional para el modelo de clasificación que para los ejemplos [Langley, 1996]. Un prototipo a su vez puede ser la representación conjunta de uno o varios ejemplos. Estos se corresponden con la transformación de los documentos en vectores de características, tal como se introdujo en la sección 2.3.3, típicamente combinado con el esquema de ponderación por frecuencias de aparición. Habiendo generado un conjunto de prototipos

a partir de una serie de documentos de entrenamiento, la clasificación de un nuevo documento se puede llevar a cabo mediante un simple proceso en dos pasos: primero se calcula la similitud del nuevo documento con cada uno de los prototipos y a continuación se utilizan estas valoraciones de similitud para determinar las categorías a las que pertenece el nuevo documento.

Hay que puntualizar que, en clasificación de textos, las medidas de similitud se aplican con más frecuencia que las medidas de distancias. Sin embargo, dichas distancias también pueden ser interpretadas como una evaluación de las diferencias entre dos elementos y, por lo tanto, están relacionadas inversamente con la similitud. Por este motivo, de ahora en adelante, cuando nos refiramos a medidas de similitud, podremos estar refiriéndonos también a medidas de distancia aplicando alguna función de transformación.

Esta breve descripción sugiere que hay tres componentes principales en cualquier clasificador basado en ejemplos: un medio de generación de prototipos a partir de los ejemplos de entrenamiento, una medida de similitud que valore el grado de semejanza entre dos vectores, y un esquema de clasificación para determinar la categoría de un nuevo documento basándose en las similitudes con los prototipos. Pasamos a describir con más detalle cada uno de ellos:

- **Generación de prototipos**. Los clasificadores basados en ejemplos varían enormemente en la forma de generar y almacenar la información representativa de cada categoría, ya que los prototipos pueden ser representaciones reales extraídas directamente de los ejemplos del conjunto de entrenamiento, o también se pueden construir artificialmente.

  En un extremo, cada ejemplo de entrenamiento puede concebirse como un prototipo, y sin ningún tipo de preproceso o paso de aprendizaje, cada documento representado como vector se puede almacenar para luego ser comparado con los nuevos ejemplos. Típicamente, esto conduce a métodos de la familia de los vecinos más cercanos, que veremos en próximas secciones. En el otro extremo, cada categoría puede ser representada por un único prototipo, calculando la media de todos los documentos de entrenamiento asignados a él. Y en medio de estos extremos, podemos encontrar técnicas que usan varios prototipos para representar cada categoría, y también técnicas que condensan y reducen el conjunto de ejemplos de entrenamiento para representar las categorías.

  El método de cálculo de medias de prototipos resulta eficiente y elegante [Langley, 1996] y sólo ocasiona un ligero incremento en el coste de aprendizaje respecto a la representación individual de documentos. Sin embargo, en su sencillez radica su debilidad, puesto que es casi imposible compendiar una gran cantidad de documentos mediante un simple promedio de sus representaciones vectoriales.

- **Cálculo de la medida de similitud**. La medida de similitud se utiliza para determinar el grado de semejanza entre dos documentos. Para conseguir unos resultados razonables, valores grandes de dicha medida deberían corresponder generalmente con elementos pertenecientes a la misma categoría, obteniendo valores pequeños en caso contrario.

- **Esquema de clasificación**. La aproximación más sencilla consiste en buscar el vecino más cercano de entre todos los vectores representativos de las categorías. Esto se puede

realizar por fuerza bruta, calculando todas las distancias entre el vector del nuevo documento y cada uno de los prototipos del conjunto de entrenamiento, para escoger a continuación como más probable la clase de aquel prototipo cuya distancia sea mínima. Sin embargo, en la práctica esto puede resultar poco aconsejable, ya que el número de comparaciones a realizar puede ser muy alto debido al número de muestras disponibles y la alta dimensionalidad de los vectores. Además, este método puede ser muy sensible al ruido, pues es posible que el elemento más cercano no se encuentre bien clasificado o, simplemente, no sea suficientemente representativo de la categoría a la que pertenece.

Para solucionar estos problemas, se han desarrollado algoritmos que encuentran por ejemplo el vecino más cercano sin recorrer exhaustivamente todo el conjunto de entrenamiento, calculando primero un centroide representativo de cada clase.[11] También podemos encontrar métodos que utilizan no sólo uno, sino varios de los vecinos más cercanos para la elección de la categoría.[12]

Máquinas de vectores de soporte

Este método fue diseñado originalmente para resolver problemas binarios, es decir, sólo se podía tomar la decisión de escoger una de entre dos posibles categorías. Las máquinas de vectores de soporte (*Support Vector Machines - SVM*) se definen en una representación vectorial donde el objetivo es encontrar la superficie de decisión que mejor separe los ejemplos de entrenamiento de dos categorías. Este concepto teórico se puede llevar a la práctica con la única premisa de que los ejemplos de las dos clases se puedan separar linealmente sin error.

En medio de todos los posibles planos de separación de las dos clases sólo existirá un hiperplano de separación óptimo, de forma que la distancia entre el hiperplano óptimo y los ejemplos de entrenamiento más cercanos sea máxima. El objetivo de maximizar el margen tiene la intención de forzar la generalización del clasificador que se esté construyendo. Aquellos documentos o ejemplos sobre los cuales se apoya el margen máximo son los denominados vectores de soporte.

A lo largo de los años han surgido diversas aproximaciones que han permitido utilizar SVM para resolver problemas de clasificación multiclase, dividiendo el problema inicial en múltiples problemas binarios [Allwein et al., 2001; Hsu y Lin, 2002; Liu et al., 2007]. Existen principalmente dos variantes:

- Uno contra todos (one-against-all). Para cada clase se construye un clasificador que decida la pertenencia a dicha clase. Por cada nuevo documento que se quiera clasificar se tendrán que tomar tantas decisiones como clases existan. Es la solución con menor coste de entrenamiento.

[11] Este tipo de metodología la utilizan los clasificadores Rocchio, basándose en una idea que surgió hace casi medio siglo [Rocchio, 1971].

[12] Método más conocido como kNN (*k Nearest Neighbours*) [Guo et al., 2004].

- Uno contra uno (one-against-one). Se construye un clasificador para cada par de clases, y por cada nuevo documento, se decide la pertenencia a una u otra clase. Más concretamente, si k es el número de categorías, se tendrán que construir $k(k\text{-}1)/2$ clasificadores. Aunque esta cifra puede dificultar el proceso de aprendizaje, se trabaja sin embargo con menos muestras y, por lo tanto, se dispone de mayor libertad para encontrar una frontera que separe cada par de clases. Esto suele permitir obtener mejor precisión durante el proceso de clasificación.

Como se detallará más adelante (sección 3.6), SVM ofrece una alta precisión y, en consecuencia, se utiliza frecuentemente como clasificador base en gran cantidad de investigaciones [Cardoso-Cachopo y Oliveira, 2003]. Sin embargo, es muy poco escalable y presenta una complejidad elevada, lo que limita su aplicación en la resolución de problemas que implican un aprendizaje exhaustivo o que utilizan corpus de gran tamaño.

Clasificadores probabilísticos

Los clasificadores probabilísticos están basados en la idea de que existe una distribución de probabilidad para definir la pertenencia de un documento con respecto al conjunto de las posibles categorías.

En su forma más simple se asume que la presencia (o ausencia) de una característica en particular dentro de una clase no está relacionada con la presencia (o ausencia) de cualquier otra característica. De esta forma, la probabilidad de pertenencia de un nuevo documento a una categoría en particular es igual al producto de las probabilidades que tiene cada uno de sus términos de aparecer en los documentos de entrenamiento de dicha categoría [Mitchell, 1997]. Tales probabilidades son fácilmente calculables partiendo, por ejemplo, de las frecuencias de aparición de cada término dentro de los documentos [Yang y Liu, 1999; Garg y Roth, 2001].

Técnicas de redes neuronales

Utilizando conceptos de inteligencia artificial, un clasificador con redes neuronales se construye mediante una red de unidades, donde las unidades de entrada son los términos, las unidades de salida representan las categorías de interés y los pesos de las conexiones son las relaciones de dependencia entre términos y categorías.

Un método de entrenamiento de redes neuronales es la retropropagación, donde se cargan los pesos de los términos como unidades de entrada y se retrocede en caso de una clasificación incorrecta, volviendo a probar cambiando los parámetros de la red con el fin de minimizar el error.

La forma más simple de este tipo de clasificadores es el perceptrón [Widrow et al., 1990], que utiliza un sistema lineal compuesto simplemente por dos capas, una de entrada y otra de salida. Además se han desarrollado también clasificadores no lineales [Ruiz y Srinivasan,

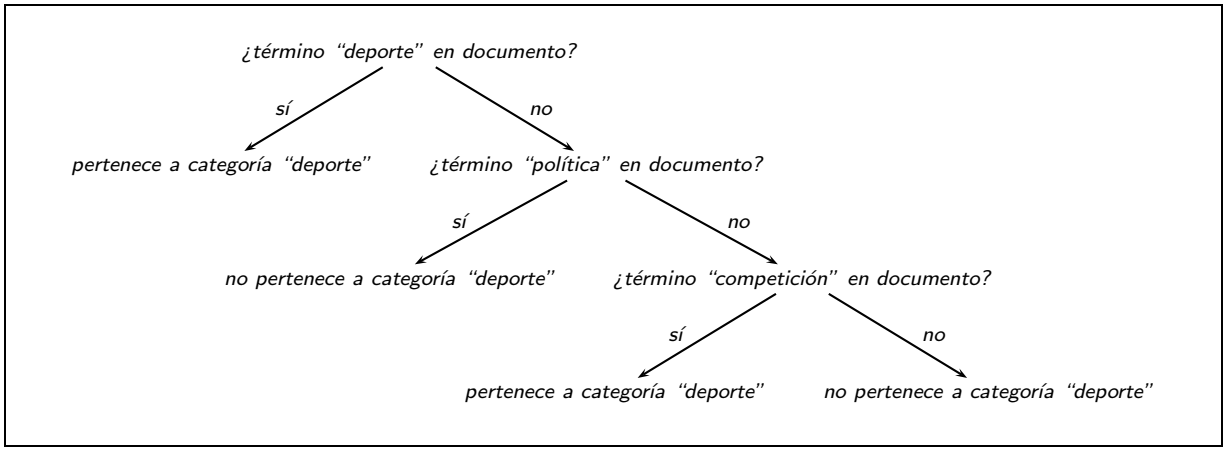


Figura 2.5: Un árbol de decisión simple. Las ramas se utilizan para tomar decisiones binarias dependiendo de si un determinado término aparece o no dentro del documento a clasificar. Los nodos hoja indican si el documento pertenece a la categoría "deporte" o no. Para ranking de categorías, se debe utilizar una estructura un poco más compleja: después de preguntar "¿El documento contiene el término?", el peso del término en el vector documento se puede usar para decidir qué rama escoger.

2002], que disponen de una o más capas de unidades adicionales entre las unidades de entrada y de salida, las cuales representan interacciones de orden muy alto entre los términos que la red es capaz de aprender. Sin embargo, en estudios previos, este diseño no llegó a ofrecer mejoras [Schütze et al., 1995] o éstas fueron mínimas [Wiener et al., 1995].

Árboles de decisión

Se plantea un sistema de clasificación mediante un árbol cuyos nodos internos representan los términos del vocabulario, las ramas que salen de ellos son etiquetadas con las pruebas sobre el peso que el término tiene en el documento y las hojas son etiquetadas con las categorías. Se puede ver un ejemplo sencillo diseñado con este planteamiento en la figura 2.5, donde observamos un clasificador binario en el que cada nodo interior se etiqueta con un término y cada rama define una comprobación sobre ese término para decidir si se debe seguir por dicha rama o no. El vector del documento recorre el árbol de decisión hasta llegar a los nodos hoja, que representan la exclusión o pertenencia a la categoría.

Hasta ahora, los clasificadores descritos eran cuantitativos por naturaleza, lo que quiere decir que la hipótesis de clasificación está basada en la aproximación a un valor real o a una función de probabilidad. Los clasificadores de árboles de decisión son diferentes, puesto que la hipótesis de clasificación se basa en una representación simbólica, construida a partir de las muestras de entrenamiento.

En resumen, clasificar un documento mediante un árbol de decisión significa recorrer recursivamente el árbol decidiendo para cada nodo interior qué rama se debería tomar. Las decisiones se basan en la representación del documento partiendo de los vectores de términos y de las reglas de decisión. El proceso de clasificación se detiene cuando se alcanza un nodo

Universitat d'Alacant
Universidad de Alicante

hoja, momento en el que se asignará la clase con la que dicho nodo se ha etiquetado.

2.4. Aplicaciones de la categorización de textos

Quizás el ámbito de aplicación más difundido para la categorización de textos es la organización de documentos en bibliotecas digitales y bases de datos bibliográficas,[13] pero podemos encontrar muchas otras aplicaciones como, por ejemplo, el control de spam, que también se encuentra entre las más destacables. En estos momentos, aplicaciones como el ya mencionado *Spamassassin* o *Mail Washer*[14] son capaces de filtrar el spam basándose en una serie de reglas, como hacían las primeras aplicaciones de categorización, utilizando aproximaciones basadas en ingeniería del conocimiento, tal y como se expuso en la sección 2.2.1. También hay otras aplicaciones, quizás no tan evidentes, como el control del acceso a Internet en colegios, o el control parental para filtrar páginas que no se consideren apropiadas.

Existe un campo de aplicación que no se puede dejar de lado: la búsqueda de información. La mayoría de los buscadores indexan la información por palabras clave, por lo que la búsqueda se limita a mostrar un listado con aquellos enlaces en los que se han encontrado ciertas palabras clave proporcionadas por el usuario. Muchas veces los resultados se complementan con una presentación ordenada por indicadores de cercanía que se basan en diferentes métricas, como por ejemplo en la cantidad de visitas de dichas páginas. Pero a veces resulta insuficiente, y por ello, algunos buscadores utilizan la clasificación de textos para guiar las búsquedas mostrando enlaces a documentos relacionados.

En otro orden de ideas, numerosos expertos han invertido tiempo y esfuerzo en diseñar técnicas de clasificación que puedan ser aplicadas en cada una de las etapas de la minería de datos, un campo de investigación emergente que se especializa en descubrir y extraer información relevante de forma automática a partir de los documentos y servicios disponibles en soporte informático [Kosala y Blockeel, 2000]. La implementación de dichas técnicas ha optimizado el proceso de recolección, clasificación y análisis de los datos en Internet con el fin de obtener conocimiento que puede ser útil, por ejemplo, para la prevención y tratamiento de enfermedades. Muchas de estas técnicas se apoyan en teorías matemáticas, estadísticas y computacionales para resolver sus problemas, permitiendo que puedan ser abordados de forma automática sin necesidad de disponer de un conocimiento lingüístico. En la figura 2.6 se observa gráficamente el proceso de extracción de información utilizando clasificación mediante aprendizaje supervisado.

Como se observa, la categorización de textos puede resultar útil en muchos ámbitos. A continuación procederemos a enumerar algunas de las principales aplicaciones que nos podemos encontrar en el mundo real, referenciando las investigaciones que avalan los resultados

[13] Entre algunas de las más conocidas bases de datos que emplean clasificación automática para organizar sus documentos podemos encontrar por ejemplo PubMed (http://www.ncbi.nlm.nih.gov/pubmed) y Eurovoc (http://europa.eu/eurovoc).

[14] http://www.mailwasher.net

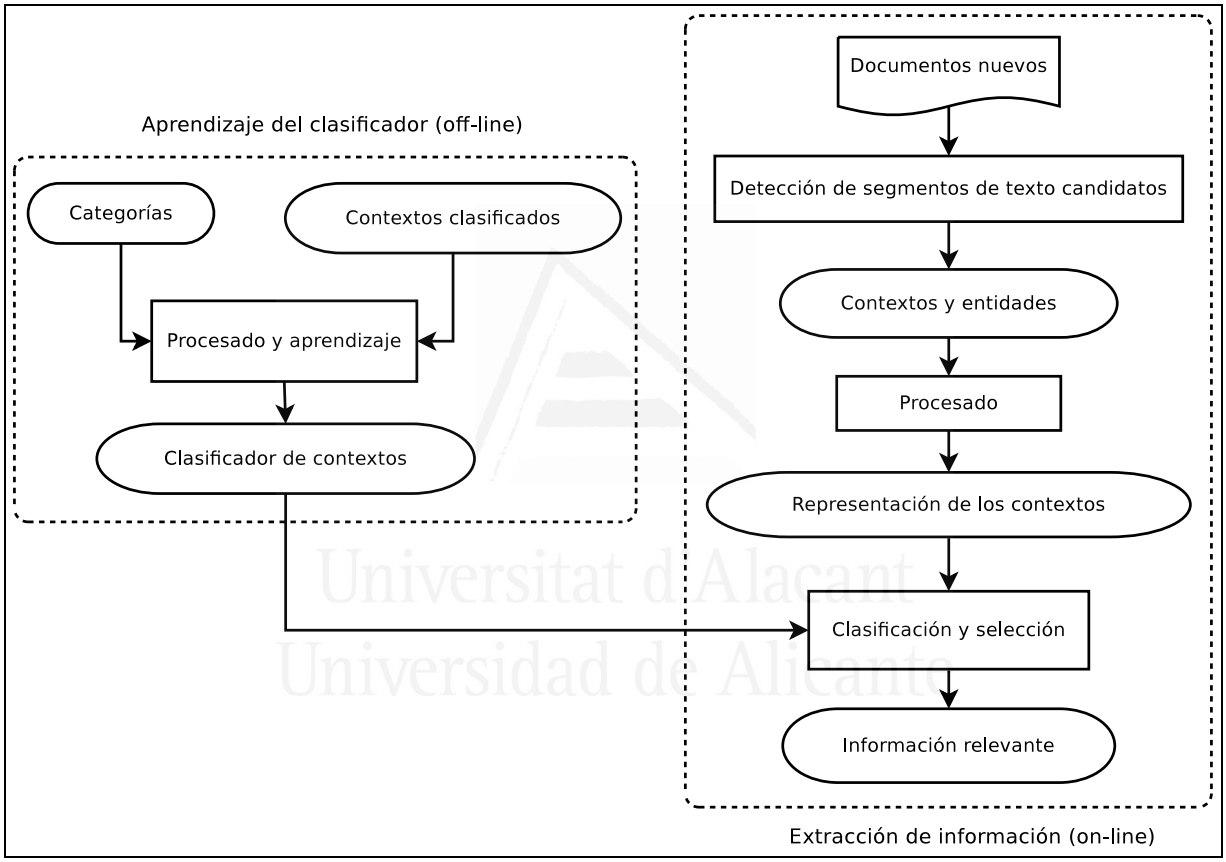


Figura 2.6: Arquitectura de extracción de información basada en clasificadores [Valero, 2005].

Universitat d'Alacant
Universidad de Alicante

obtenidos para cada uso particular.

2.4.1. Filtrado de textos

El filtrado de textos es la actividad de clasificar un flujo de documentos entrantes por vía asincrónica de un productor a un consumidor de información [Belkin y Croft, 1992]. Para llevar a cabo dicho filtrado sólo se necesita un clasificador binario que asigne a cada documento una de entre dos posibles categorías: relevante o irrelevante. Aunque en principio esta aplicación se reduce a un caso de etiquetado simple, adicionalmente se podría realizar clasificación dentro de ciertas categorías de interés, llevándonos a un proceso conocido como encaminamiento (*routing*) [Guthrie et al., 1999]. También nos podemos encontrar con un filtrado adaptativo (*adaptative filtering*) [Liddy et al., 1994] donde el aprendizaje se realiza para diferentes usuarios, manteniendo perfiles de cada uno y ampliando así las posibilidades de personalización.

Como se ha comentado en secciones anteriores, una de las aplicaciones más habituales del filtrado de textos es la de los sistemas de detección de correo no deseado (spam). En [Drucker et al., 1999] podemos encontrar una descripción detallada de cómo utilizar máquinas de vectores de soporte para dicho propósito. En [Androutsopoulos et al., 2000] podemos observar los resultados obtenidos al filtrar correo mediante clasificadores probabilísticos.

También podemos utilizar filtrado para restringir el acceso a ciertas páginas web. Por ejemplo, muchos buscadores limitan por defecto los resultados de sus búsquedas para que no se muestren contenidos no aptos para menores. Entre otros, se puede consultar [Gómez et al., 2004], donde se describe el proceso de desarrollo de un clasificador inductivo adaptando el diseño para filtrar contenidos disponibles en Internet.

Universidad de Alicante

2.4.2. Clasificación de noticias

Si observamos cualquier periódico, tanto en formato electrónico como en papel, veremos que se encuentra estructurado en diferentes secciones. De esta forma se pretende que los usuarios interesados en un determinado tema puedan acceder directamente sólo a la información que les resulte más atractiva. Mientras que la distribución habitual engloba las clásicas secciones periodísticas (política, economía, deportes, internacional, etc.), también nos podemos encontrar otros intereses mucho más concretas como por ejemplo fútbol, baloncesto, videojuegos, libros, cine, teatro y un largo etcétera. Esto lleva a intentar automatizar un proceso de etiquetado en categorías para ofrecer un amplio abanico de secciones, dando la posibilidad al usuario de seleccionar sólo aquellas noticias que le sean relevantes.

Hay que tener en cuenta que el número de noticias que existe en Internet en la actualidad es cada vez mayor, lo que dificulta (o imposibilita) su indexación y clasificación manual. Con esta motivación y gracias a que existen multitud de colecciones de documentos ya etiquetados, se han realizado numerosos estudios hasta la fecha [Maria y Silva, 2000; Wermter y Hung,

2002; Mittermayer y Knolmayer, 2006]. En algunos casos se perfeccionaron dichas aplicaciones para conseguir presentar soluciones adaptativas al perfil del usuario [Billsus y Pazzani, 1999; Dörfler et al., 2001]. Tal vez los sistemas más conocidos utilizados en el mundo real son Google News[15] y Newsblaster.[16] El primero, además de clasificar las noticias por temas, agrupa noticias de diferentes medios que tratan del mismo tema, facilitando el acceso para el lector.

2.4.3. Categorización jerárquica de páginas web

El objetivo de esta categorización es incrementar la funcionalidad de los motores de búsqueda, facilitando la navegación por Internet al agrupar los enlaces por categorías jerárquicas, restringiendo de esta forma la colección de documentos sobre los que se aplican las consultas. Al hacerlo de forma automática, obtenemos la obvia ventaja de la movilidad de las categorías, y al tratarse de páginas web, contamos con dos características que nos llevan a conseguir clasificadores eficientes sin la ayuda de expertos humanos:

- La naturaleza hipertextual de los documentos, por lo que la clasificación varía respecto de lo interesante de las páginas referenciadas. En [Yang et al., 2002] se ofrece una comparativa muy completa de diferentes sistemas que aprovechan esta característica. Además, también se pueden encontrar otros estudios posteriores que refuerzan dichos resultados [Calado et al., 2003; Lu y McKinley, 2003; Fisher y Everson, 2003; Karamon et al., 2007].

- La estructura jerárquica del conjunto de categorías, que permite dividir el problema en otros de menor dimensión. Algunas de las investigaciones que aprovechan esta peculiaridad son [Weigend et al., 1999; Dumais y Chen, 2000; Ruiz y Srinivasan, 2002]. Además, en [Ceci y Malerba, 2007] podemos encontrar un completo estudio sobre el tema.

2.4.4. Identificación del idioma

Internet contiene una gran colección de documentos escritos en muchos idiomas diferentes y, por lo tanto, puede resultar muy interesante conseguir delimitar las búsquedas para mostrar sólo aquellos documentos en un idioma especificado. Entre alguno de los intentos por lograr dicho objetivo podemos encontrar [Cavnar y Trenkle, 1994], donde se presentó un algoritmo que posteriormente dio lugar al desarrollo de TextCat,[17] un sistema de identificación de lenguajes que ofrece muy buenos resultados y que se sigue utilizando en la actualidad.

[15] http://news.google.es
[16] http://newsblaster.cs.columbia.edu
[17] http://www.let.rug.nl/~vannoord/TextCat

2.4.5. Desambiguación del sentido de las palabras

Dada la ocurrencia de una palabra ambigua (o polisémica) dentro de un texto, se realiza una clasificación mediante etiquetado simple con el objetivo de encontrar el significado más adecuado de dicha palabra. Entre los numerosos estudios que utilizan la clasificación de textos para la desambiguación de palabras, podemos encontrar [Gale et al., 1993; Roth, 1998; Escudero et al., 2000; Carpuat et al., 2004].

Ejemplos interesantes de uso, que están contenidos dentro de los problemas de lingüística computacional, son la corrección ortográfica sensible al contexto [Golding y Schabes, 1996; Roth, 1998], las sugerencias de redacción [Kilgarriff, 1997] y la deducción del significado de las palabras en las traducciones [Li y Li, 2004].

2.4.6. Etiquetado de entidades

Este campo consiste en identificar qué entidades u objetos se mencionan en un texto. Una entidad es una palabra o una secuencia de palabras que pueden ser clasificadas dentro de un determinado tipo, como podrían ser el nombre de una persona o de una organización. También puede utilizarse para ámbitos específicos como por ejemplo, en el campo de la medicina, donde las entidades podrían ser nombres de moléculas, genes, proteínas, nombres de enfermedades o de tratamientos, nombres de órganos, etc. [Roa, 2008]. Esta tarea parece fácil cuando se hace de forma manual, pero si lo intentamos automatizar puede resultar difícil por la ambigüedad de las palabras y porque no todos los documentos son uniformes. La mayoría de técnicas usan patrones, carácter a carácter o palabra a palabra, para identificar las entidades. Aunque los patrones se pueden definir manualmente, hoy en día es más común utilizar patrones que son aprendidos a partir de un conjunto de ejemplos [Nadeau y Sekine, 2007].

Universitat d'Alacant
Universidad de Alicante

2.4.7. Clasificación de preguntas para sistemas de búsqueda de respuestas

Los sistemas de búsqueda de respuestas (*Question Answering* - QA) tienen como propósito contestar preguntas formuladas en lenguaje natural, proporcionando una simple contestación, en vez de devolver el texto completo de los documentos donde pueda aparecer la respuesta.

Un paso importante en QA consiste en clasificar la pregunta para poder anticipar el tipo de respuesta esperada [Tomás, 2009; Huang et al., 2009] . Por ejemplo, la pregunta "¿Quién descubrió los rayos X?" debería ser clasificada dentro del tipo de personas o individuos. Esta información reducirá la búsqueda de la frase que contiene la respuesta correcta y también nos puede ayudar en la ponderación de las diversas respuestas que se puedan considerar como candidatas.

La clasificación de preguntas presenta dificultades añadidas a un sistema tradicional de clasificación de textos [Tomás, 2009] y, por lo tanto, no es en absoluto un problema de solución

trivial:

- Se dispone de menor cantidad de texto en una simple pregunta que en un documento completo, siendo más difícil evaluar la similitud entre la pregunta y las diferentes categorías, pues dispondremos de menor texto coincidente en los ejemplos de entrenamiento.
- No podemos ponderar las características extraídas del texto dependiendo de la posición donde las encontremos, ni dependiendo de la cantidad de veces que aparecen, ya que sólo se nos proporciona una pregunta formulada mediante unas pocas palabras. No disponemos de apartados especiales tales como el título, el asunto o el autor. Además, el texto a evaluar es tan escueto que las mismas palabras no suelen repetirse. Por lo tanto, puede resultar necesario expandir la pregunta para ampliarla usando estrategias tales como la inclusión de sinónimos.
- El uso de palabras de parada está muy limitado, pues podríamos eliminar de la pregunta pronombres interrogativos clave, tales como "quién", "cuándo" o "por qué".
- Es difícil encontrar una sola solución que funcione bien en entornos multilingües, debido a las variaciones típicas del lenguaje que se deben considerar. Existen soluciones que utilizan corpus paralelos [Tomás et al., 2005], mientras que otros autores proponen traducir al inglés la consulta del usuario [Cumbreras et al., 2005]. Aún si elegimos la opción aparentemente más sencilla, y limitamos el problema a la lengua inglesa, nos encontraremos que todavía resulta difícil intentar clasificar preguntas del tipo "¿Cuál...?" o "¿Qué...?". Consideremos por ejemplo la consulta "¿Cuál es la capital de España?", que es del tipo lugar, mientras que "¿Qué es la escala pH?" es del tipo definición. Aunque ambos ejemplos deben ser etiquetados de forma diferente, son traducidos al inglés mediante la misma partícula interrogativa ("What...?").
- La portabilidad entre varios dominios también se complica. Muchas de las herramientas y recursos que se emplean se encuentran optimizados para un dominio específico y un tipo de datos determinado. En la bibliografía encontramos por ejemplo que la mayoría de las evaluaciones en la clasificación de preguntas se han realizado sobre los corpus proporcionados en las competiciones organizadas por las conferencias internacionales TREC, CLEF y NTCIR. Se dispone de pocos casos (por ejemplo [Beitzel et al., 2007; Sundblad, 2007]) que proporcionan comparaciones utilizando corpus de usuarios reales.

Si revisamos la bibliografía, observaremos que la clasificación de preguntas tiene una gran influencia dentro de la calidad de la respuesta final ofrecida por los sistemas de QA [Moldovan et al., 2003] y, por ello, ha recibido una significativa atención durante la última década [Loni, 2011].

Universitat d'Alacant
Universidad de Alicante

Capítulo 3

Trabajos previos

A continuación revisaremos los principales trabajos desarrollados en el ámbito de esta tesis doctoral.

En las primeras secciones de este capítulo pretendemos mostrar algunas de las técnicas existentes para la obtención de características y los correspondientes pesos que indican su relevancia. Comprobaremos que dicho proceso se puede automatizar partiendo del contenido textual de corpus específicos haciendo uso de sencillas técnicas de procesamiento a nivel léxico o sintáctico, que obtengan los diversos tokens, junto con ciertos elementos estadísticos. Veremos por ejemplo que podemos considerar simples palabras para representar cada documento y que podemos valorarlas contabilizando las frecuencias de aparición en la colección de documentos de entrenamiento. Pero la extracción automática de las expresiones nos llevará a obtener vectores de características de muchas dimensiones, hecho que ocasionará inevitablemente que el proceso de clasificación sea menos eficiente. Debido a esto, analizaremos también algunas técnicas de reducción de dimensionalidad que permitan por ejemplo prescindir de términos innecesarios o de aquellos que tengan una ponderación muy baja.

Una vez dispongamos de los vectores de características que representan cada documento, podremos aplicar alguna función para calcular la similitud entre dos vectores. Pero aunque existe un amplio abanico de posibilidades para abordar este problema [Moffat y Zobel, 1998], sólo trataremos algunas de ellas y no lo haremos en detalle, pues en realidad todas parten de los mismos elementos de información [Van Rijsbergen, 1979].

Detallaremos también las principales técnicas o algoritmos de aprendizaje automático que permitirán utilizar muestras ya clasificadas para obtener una lista ordenada de posibles categorías presentando un texto no visto anteriormente. Veremos cómo valorar las semejanzas observadas entre el conjunto de características generado durante la fase de entrenamiento y las características extraídas de los nuevos documentos.

Al final de este capítulo analizaremos diversos algoritmos que intentan optimizar los procesos de aprendizaje y clasificación. La mayoría de las soluciones pretenden minimizar la cantidad de recursos requeridos y mejorar la precisión en los resultados. A su vez se intenta

reducir el tiempo necesario para la construcción del clasificador y el tiempo requerido para obtener respuestas una vez ya construido. En este aspecto, veremos soluciones que intentan condensar o representar cada categoría mediante un solo vector, o soluciones que persiguen un modelado individual para cada temática, que permita separar el conjunto de muestras en dos zonas para indicar la pertenencia o no pertenencia a la correspondiente categoría.

3.1. Representación de documentos

Intuitivamente, parece razonable asumir que una representación más compleja obtenida de niveles más altos de análisis del texto, desembocará en clasificadores más eficaces. Esto podría llegar a ser cierto si dispusiéramos de una cantidad infinita de texto reflejando información sobre las características que deben ser extraídas. Sin embargo, en problemas reales, el número de documentos disponibles para el análisis del texto es limitado y, si el conjunto de documentos de entrenamiento es pequeño, es más difícil inducir un clasificador que ofrezca una alta precisión en sus resultados.

Pero aún disponiendo de una gran cantidad de información, nos encontraremos con el inconveniente de que cuanto más complejas sean las definiciones de términos, mayor será la dimensionalidad en el espacio de características.[1] Además, tal como demuestra la investigación en recuperación de información, existen gran variedad de representaciones textuales y, a pesar de estar basadas en criterios diferentes, muchas de ellas han conducido a una efectividad muy similar en los resultados obtenidos. En [Lewis, 1992b] se describe este hecho como la paradoja de la igualdad en la eficiencia.

No obstante, es importante destacar que estas conclusiones se basan en resultados obtenidos principalmente de documentos escritos en inglés. Para otros idiomas, un análisis lingüístico más profundo puede resultar beneficioso. En alemán, por ejemplo, existen muchas palabras compuestas, tales como "Textklassifikationsverfahren", que pueden causar un incremento importante en el número de características. En este caso, técnicas que reconozcan dichas composiciones y las separen en partes más pequeñas, podrían solucionar el problema. El beneficio que se obtiene de un análisis lingüístico tiene como contrapartida una alta dependencia del lenguaje y del dominio de aplicación.

De los planteamientos realizados, podemos deducir que para escoger un nivel de análisis de texto adecuado en el que basar la definición de los términos siempre se debe mantener un equilibrio entre expresividad semántica y complejidad representacional. Esta elección tendrá un alto impacto en la habilidad del algoritmo de aprendizaje para generalizar. Por este motivo, definiciones simples de términos son más comunes en la clasificación de textos [Apté et al., 1994a]. Típicamente, esto implica análisis a nivel léxico, aunque también es común incorporar cierta información sintáctica. A continuación procedemos a describir tres definiciones de términos usadas ampliamente que básicamente explotan algunos hechos estadísticos sobre el

[1] Tal como se comentó en la sección 2.3.3, el problema de obtener demasiadas características al procesar los documentos de entrenamiento se conoce comúnmente como *maldición de la dimensionalidad*.

texto [Neumann y Schmeier, 1999]. La primera aproximación depende de n-gramas de letras y por lo tanto, corresponde a un nivel de análisis grafémico. Las otras dos soluciones (palabras individuales y frases) se incluyen dentro del análisis léxico, aunque para construir las frases se necesite aplicar algunas técnicas básicas de análisis sintáctico.

3.1.1. N-gramas de letras

Se trata de secuencias de n letras contiguas [Cavnar y Trenkle, 1994], aunque a veces el término "n-grama" se utiliza también para referirse a secuencias de palabras de longitud n [Furnkranz, 1998]. Nosotros nos referiremos a estos términos compuestos de múltiples palabras como frases. En el análisis de n-gramas de letras, se utilizan habitualmente bigramas ($n = 2$), trigramas ($n = 3$) o cuatrigramas ($n = 4$), aunque también se pueden encontrar muchos otros tipos. Por ejemplo, de la palabra "casa" se podrían extraer los trigramas "cas" y "asa". Incluso también es posible incluir a la izquierda o a la derecha de una palabra un espacio en blanco al generar los n-gramas. En este caso, la palabra "casa" tendría adicionalmente los trigramas " ca" y "sa ". Teniendo en consideración sólo las 26 letras del alfabeto inglés, podríamos obtener $26^3 = 17576$ trigramas y $26^4 = 456976$ cuatrigramas diferentes, aunque sabemos que no todas las combinaciones se presentarán en el idioma. A medida que incrementemos el valor de n, obtendremos n-gramas más largos y, por lo tanto, capaces de abarcar palabras enteras. Por ello, valores altos de n harán que el vector resultante de características sea más legible y, por lo tanto conseguirán representar mejor el contenido semántico de los documentos originales [Tauritz et al., 2000]. Sin embargo, hay que tener en cuenta ciertos márgenes, pues n-gramas de 3 letras no proporcionan suficiente información y valores de n superiores a dicha cifra nos conducen a números muy elevados de posibles términos [Teufel y Schmidt, 1988].

Es importante destacar que la ventaja principal del uso de n-gramas es que el conjunto posible de términos es fijo y se conoce por adelantado. Por ello, la extracción de n-gramas es un proceso relativamente sencillo e independiente del lenguaje. Además, los n-gramas son bastante robustos cuando se presentan variaciones morfológicas y errores de ortografía. Como contrapartida, obtendremos una representación final que será difícil de analizar por humanos.

3.1.2. Palabras

Un enfoque muy común es utilizar cada palabra como un token, tal cual, sin ninguna transformación. Investigaciones previas demuestran que simples palabras pueden utilizarse como características con un alto grado de eficacia [Dumais et al., 1998]. Por ello resulta habitual representar los documentos como un conjunto de términos (lo que se conoce como "bolsa de palabras") sin tener en cuenta el orden en el que aparecen en el texto [Lewis y Ringuette, 1994].

Obviamente, en su versión básica, esta definición de términos es independiente del lenguaje y muy eficiente computacionalmente hablando. Sin embargo, presenta la desventaja de

que cada inflexión de una palabra se considera como una característica diferente y, por lo tanto, puede llegar a obtenerse un conjunto del vocabulario innecesariamente grande. Además, variaciones morfológicas de una palabra no son reconocidas como similares. Aunque una sencilla solución puede consistir simplemente en reducir cada palabra a su raíz, hay que aclarar que esto dificulta la extracción de términos, pues la hace dependiente del lenguaje y menos eficiente a nivel computacional.

3.1.3. N-gramas de palabras

Los n-gramas de palabras o frases son combinaciones de tokens. El uso de frases está justificado por el hecho de que muchos términos están formados por varias palabras (sobre todo en inglés). Por ejemplo, nos podemos encontrar con "data mining" o "information retrieval". Al igual que los n-gramas de letras, podemos elegir la cantidad máxima de palabras que serán consideradas como una única característica. Podemos hablar de unigramas, bigramas, trigramas, etc. Habitualmente se tendrá en cuenta el idioma y el dominio de aplicación y sólo se utilizarán frases específicas con un número reducido de palabras, porque de otro modo, el número de posibles características se incrementaría drásticamente.

Los términos formados por varias palabras se pueden obtener observando secuencias de palabras que aparecen con frecuencia [Cohen y Singer, 1996], o se pueden detectar mediante técnicas de procesado del lenguaje natural, método que también es conocido como indexado sintáctico de frases [Furnkranz, 1998]. Otro enfoque para la identificación de frases es la aportación manual de un conjunto de posibles frases específicas de cada dominio [Sahami, 1998b].

Los resultados obtenidos con el uso de frases difieren mucho los unos de los otros: algunos investigadores mostraron una importante mejora en la precisión de los clasificadores usando frases [Cohen y Singer, 1996], mientras que otros no consiguieron construir mejores clasificadores que los que usaban simples palabras [Dumais et al., 1998]. Algunos autores han argumentado que estas discrepancias pueden estar justificadas por los modelos utilizados durante el aprendizaje. Por ejemplo, aquellos modelos que no tienen en cuenta las co-ocurrencias entre palabras pueden beneficiarse del uso de frases, puesto que las frases incluyen intrínsecamente tal información. Por otro lado, los modelos capaces de guardar las co-ocurrencias no obtienen ningún beneficio por el uso de frases, e incluso pueden resultar perjudicados al tener que estimar parámetros adicionales por el uso de tales términos [Sahami, 1998b].

3.1.4. Otras características

También nos podemos encontrar otro tipo de características que no dependen tanto del análisis lingüístico del texto y cuyo uso es menos habitual. Por ejemplo, puede resultar útil identificar y guardar los autores y las fuentes de los documentos, o ciertas entidades tales como nombres propios, siglas, o vocabulario específico del corpus. Podemos tomar como caso claro

la categorización de publicaciones científicas en el ámbito médico, donde nos encontramos con revistas especializadas en alguna de tantas áreas en las que se encuentra dividida la medicina y con autores que suelen realizar contribuciones sólo dentro de su especialidad. Por lo tanto, futuros artículos en la misma revista o del mismo autor tienen una alta probabilidad de versar del mismo tema.

3.2. Reducción de la dimensionalidad

Generalmente la expresión "reducción de dimensionalidad" se utiliza para agrupar cualquier procedimiento cuyo objetivo sea controlar la dimensionalidad de un espacio de características. Esto incluye técnicas de selección que intentan encontrar un subconjunto adecuado de características mediante la inclusión o exclusión de algunos términos en particular para formar el conjunto final. A la hora de representar nuevos documentos, esto nos permitirá descartar directamente muchos términos, pues sólo se deberán tener en cuenta aquellos existentes en el vocabulario. También es muy común utilizar técnicas basadas en la extracción de características o en el ajuste de parámetros [Schütze et al., 1995], consistentes en construir nuevas características tomando combinaciones y transformaciones del conjunto original. Este último tipo de técnicas en realidad reemplazan características existentes por otras nuevas para reducir la dimensionalidad del vocabulario, realizando una asociación entre los términos originales y los generados, no sólo para los documentos de entrenamiento, sino para nuevos documentos.

A continuación procedemos a enumerar algunas de dichas técnicas. Comenzaremos por explicar el concepto de *stopwords*, que sugiere el uso de una lista fija de expresiones poco relevantes (generalmente establecida a mano) que conviene descartar sin tener en cuenta ningún tipo de análisis. Después justificaremos la consideración de un umbral mínimo (o máximo) de apariciones por debajo (o por encima) del cual las palabras se consideran poco significativas. Concluiremos la sección explicando otras formas de selección de términos muy utilizadas.

3.2.1. Métodos de selección de características

Las técnicas de selección de términos para reducción de dimensionalidad toman como entrada un conjunto de características y proporcionan como salida un subconjunto de dichas características, que son relevantes para discriminar entre las categorías.[2] Idealmente, sería interesante obtener un subconjunto entre todos los posibles subconjuntos que permitan a un algoritmo en particular inducir la mejor hipótesis posible respecto a una medida de efectividad [Kohavi y John, 1997]. Por ello, la selección de características puede ser abordada como un problema de optimización donde el espacio de búsqueda corresponde a todas las posibles combinaciones de subconjuntos (si m es el número total de características, se obtendrán

[2]Se puede consultar [Dash y Liu, 1997] para obtener un listado detallado de técnicas de selección de características utilizadas en tareas de clasificación.

un total de 2^m posibles candidatos), ya que una característica podrá aparecer o no en el subconjunto. Obviamente, esto provoca una búsqueda exhaustiva intratable, puesto que el número de características es muy grande. Por lo tanto, es muy recomendable que la selección de características sea guiada por heurísticas [Schürmann, 1996].

El enfoque más simple de filtrado trata cada característica de forma independiente. A cada una se asigna una puntuación para decidir si incluirla o excluirla en el conjunto del vocabulario. El vocabulario final se establece seleccionando aquellas características que tengan una puntuación por encima o por debajo de un determinado umbral o seleccionando aquellas que obtengan la mejor puntuación. Pero determinar el umbral adecuado o el número de características a seleccionar es todavía un problema complejo.

Comenzaremos explicando un enfoque lingüístico frecuentemente utilizado conocido como la eliminación de palabras de parada (*stopwords*). Después expondremos diversas medidas numéricas para evaluar el peso de los términos respecto a su poder discriminativo entre clases. Habitualmente, todas las medidas se basan en las frecuencias que identifican la aparición de los términos en los documentos, en las categorías, o en toda la colección de documentos.

Eliminación de palabras de parada

Si analizamos cualquier idioma, podemos observar a menudo que hay palabras que aparecen en todos los documentos sin importar el tema del que puedan tratar. Estas palabras con una alta frecuencia de aparición son conocidas como palabras de parada (*stopwords* [Van Rijsbergen, 1979]). Habitualmente, son palabras tales como artículos, preposiciones, conjunciones y pronombres, que proporcionan simplemente estructura al lenguaje pero no aportan contenido [Sahami, 1998a]. Puesto que las stopwords carecen de carácter discriminativo, parece muy razonable eliminarlas del conjunto de posibles términos válidos y como resultado, reducir la dimensionalidad del espacio de características asociado. Se debe aclarar que en el caso de un vocabulario formado por n-gramas de letras, la eliminación de palabras de parada debería tener lugar antes del paso de extracción de los n-gramas.

Hay dos formas de obtener una lista de stopwords. Un enfoque muy utilizado es establecer manualmente dicha lista (habitualmente, estas listas contienen varios cientos de palabras). Para el idioma inglés se encuentran numerosas propuestas en la literatura científica [Van Rijsbergen, 1979; Fox, 1992]. En algunos casos, también puede resultar útil proporcionar listas específicas para el dominio de aplicación, además de la lista general, o incluso para reemplazar la lista genérica.

Una segunda posibilidad para eliminar stopwords es crear esta lista de forma automática basándose en la colección de documentos de entrenamiento. Para hacer esto, se pueden considerar como stopwords aquellas palabras cuya frecuencia de aparición esté por encima de un determinado umbral [Lang, 1995]. El umbral puede ser definido manualmente a priori, o puede calcularse partiendo del histograma de frecuencias de la distribución de los términos. Este enfoque elimina las stopwords que son específicas de la colección de documentos en consideración y, por lo tanto, sólo es útil en el dominio de aplicación para el que se ha

obtenido. De esta forma, no sólo se filtrarán palabras de parada comunes, sino que también se eliminarán algunas tales como nombres, verbos y adjetivos.

Frecuencia de términos

Una heurística de selección muy simple es eliminar todos aquellos términos cuyas frecuencias de aparición en la colección de documentos están por encima de un límite superior o por debajo de un límite inferior preestablecido [Luhn, 1958]:

La justificación de medir la importancia de una palabra mediante la frecuencia de uso está basada en el hecho de que un escritor normalmente repite ciertas palabras a medida que avanza en el texto o varía sus argumentos y a medida que elabora el contenido de un aspecto de un tema. Este medida para enfatizar algo, se puede tomar por lo tanto como un indicador de importancia.

De esta forma, se considera habitualmente que los términos que aparecen poco en una colección de documentos tienen escaso valor y pueden ser eliminados [Van Rijsbergen, 1979]. Por otro lado, se asume que los términos con una alta frecuencia de aparición son muy comunes y por lo tanto, no tienen ningún poder discriminativo. En el siguiente párrafo expondremos la Ley de Zipf, que observa este fenómeno y justifica por qué es conveniente prescindir de las expresiones que tengan dicho perfil. De forma intuitiva, ya podemos deducir que eliminar estos términos corresponde de hecho a la eliminación de palabras de parada, donde la lista se construye automáticamente partiendo de la colección de documentos, basándose en el histograma de las frecuencias de términos.

Hace más de 50 años, después de realizar estudios empíricos, Zipf observó que en una colección de documentos un pequeño número de palabras son utilizadas con mucha frecuencia, mientras que frecuentemente ocurre que un gran número de palabras son poco empleadas [Zipf, 1949]. La conclusión a la que llegó se conoce como Ley de Zipf y, muchos años después, todavía se valora como un fenómeno matemático [Sahami, 1998a]. A pesar del hecho de que los análisis de Zipf y Luhn [Luhn, 1958] se basaban originalmente en palabras, sus teorías también se pueden aplicar sobre otro tipo de representaciones, tales como n-gramas o frases [Van Rijsbergen, 1979]. Por lo tanto, para las siguientes explicaciones nos referiremos a términos de manera general en vez de únicamente palabras.

Supongamos que disponemos de todos los términos del vocabulario ordenados en orden decreciente de acuerdo con el número total de apariciones (frecuencia) en la colección de documentos y que $posicion(t)$ es la posición del término t en la lista ordenada. La Ley de Zipf establece que:

$$posicion(t) \cdot frecuencia(t) \approx constante \ . \tag{3.1}$$

El gráfico 3.1 ilustra la aplicación de Luhn de la Ley de Zipf. La frecuencia de los términos en función de su posición viene indicada por la curva hiperbólica. Las líneas verticales de-

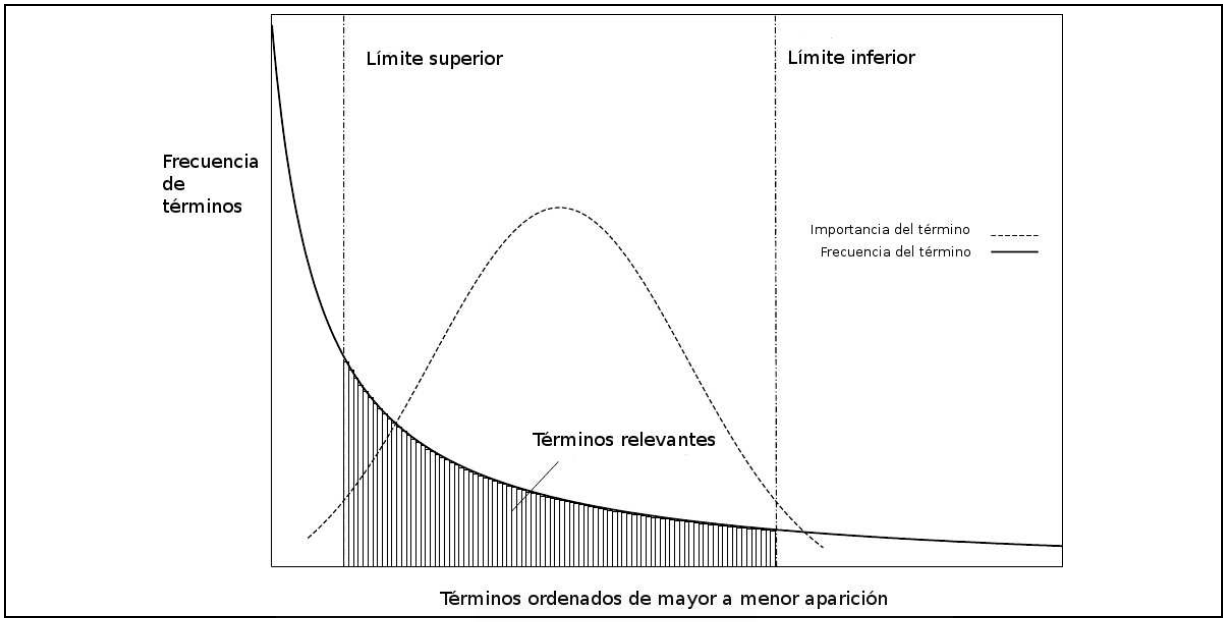


Figura 3.1: Ley de Zipf aplicada según [Luhn, 1958].

muestran posibles umbrales para las frecuencias de términos. Mientras las frecuencias altas y bajas tienen poco poder discriminativo, las palabras restantes, que tienen una frecuencia media, se consideran relevantes para discriminar entre clases. Se asume que la importancia de los términos alcanza su valor máximo entre los dos umbrales de corte, como se muestra en la campana de la figura 3.1. Eliminando aquellos términos no significativos por sus altas o bajas frecuencias, la dimensionalidad del espacio de características se puede reducir hasta un 50 % [Van Rijsbergen, 1979]. Es conveniente mencionar que la tarea de encontrar los umbrales adecuados no es trivial y todavía se están realizando estudios. Además, dichos valores dependen en gran medida de la colección de documentos y muchas veces deben ser ajustados de manera empírica.

Frecuencia en el documento (DF)

Una heurística muy simple consiste en excluir aquellos términos del vocabulario cuya frecuencia de aparición en los documentos es menor que un determinado umbral. Se ha demostrado que se puede disminuir la dimensionalidad del conjunto de términos hasta en un factor de 10 sin pérdida de información y hasta en 100 con una pérdida de información despreciable [Yang y Pedersen, 1997].

Esto se basa en la asunción de que los términos que aparecen en sólo unos pocos documentos no proporcionan información específica de una clase y a veces incluso tienden a aportar mucho ruido, como puede ser el producido por ejemplo, por los errores ortográficos. Además, el hecho de utilizar términos muy poco frecuentes no es estadísticamente fiable [Apté et al., 1994b]. Por lo tanto, eliminar estos términos no sólo mantiene el poder discriminativo, sino que además puede mejorar la efectividad del proceso de clasificación.

La eliminación de términos por frecuencia del documento es una técnica sencilla y muy efectiva y, habitualmente, se aplica antes de otros métodos de selección más sofisticados. La mayoría de los autores eliminan aquellos términos que aparecen en un conjunto muy reducido de documentos. Por ejemplo, resulta muy común descartar términos que aparecen en 3 o menos documentos [Granitzer, 2003].

Sin embargo y a pesar de su amplio uso, no deja de ser una consideración muy genérica que puede acarrear ciertos riesgos. Hay que tener en cuenta que dentro del ámbito de la recuperación de información, un término que aparece sólo en un documento puede ser especialmente útil para identificar dicho documento.

3.2.2. Métodos de extracción de características

Las técnicas de extracción de características utilizadas para reducir la dimensionalidad toman como entrada un conjunto de características derivadas de los documentos de entrenamiento y construyen un nuevo conjunto de menor tamaño tomando combinaciones y transformaciones de las características existentes, mientras se mantiene o incluso se mejora su poder discriminativo. Esta conversión se aplica antes de las fases de entrenamiento y clasificación para obtener los vectores representativos de los documentos.

El objetivo es similar a la selección de términos, sólo que la diferencia está en que ahora se busca el aumento de la efectividad en la existencia de palabras polisémicas, hominómicas y sinonímicas, fijándonos por ejemplo en las apariciones de palabras diferentes que expresen el mismo significado.[3] De esta forma pueden llegar a transformarse las expresiones de los textos originales y, por ello, hay que hacer hincapié en que las características resultantes en algunos casos serán artificiales y es posible que a simple vista no tengan correspondencia directa con cada uno de los términos que aparecen en los documentos.

A continuación expondremos algunas de las transformaciones lingüísticas más utilizadas que nos permitirán reducir la dimensionalidad: stemming, tesauros y agrupación del texto (o clustering). Se puede consultar además [Fodor, 2002; Han et al., 2004] para ampliar información sobre otros métodos más sofisticados, que debido a su complejidad no han tenido tanta aceptación en el ámbito de la clasificación de textos.

Stemming

Tal como se ha expuesto anteriormente, al utilizar el modelo de bolsa de palabras, se obtiene con frecuencia un número innecesariamente alto de características, debido a que las variantes morfológicas de las palabras no son reconocidas como similares y cada inflexión es una característica potencial. El stemming de palabras es un proceso pseudo-lingüístico que elimina sufijos para reducir las palabras a su raíz [Smeaton, 1997]. Por ejemplo, las palabras "clasificador", "clasificado" y "clasificando" se podrían reducir a su raíz "clasifica".

[3] Más información en http://es.wikipedia.org/wiki/Polisemia.

Consecuentemente, la dimensión del espacio de características puede ser reducida al asociar morfológicamente palabras similares en su raíz.

Un algoritmo de stemming ampliamente utilizado es el eliminador de sufijos desarrollado por Porter cuya primera versión data de 1980 [Porter, 1980]. Los algoritmos de stemming se basan frecuentemente en heurísticas y muchos por lo tanto reducen palabras que no son realmente similares. Por ejemplo, el stemmer de Porter reduce la palabra "is" a "i". En [Frakes y Baeza-Yates, 1992] se describen varios algoritmos de stemming y se proporciona una visión general de algunos estudios comparativos entre los distintos métodos para tareas de recuperación de información. Basándose en esta comparación, en [Sahami, 1998a] se establece que ambos métodos de representación (mediante stem y sin él) proporcionan un rendimiento muy similar. Obviamente, el stemming depende del idioma empleado, pero no del dominio de aplicación. Es relativamente sencillo especificar simples heurísticas de stemming para palabras en inglés, mientras que proporcionar reglas para stemming en otros lenguajes tales como alemán o francés, puede ser mucho más difícil. Por último, hay que recalcar que en el caso de los términos basados en n-gramas, el stemming (si se realiza), debe tener lugar antes de la etapa de extracción de términos y debe aplicarse sobre los tokens de palabras obtenidos del paso de normalización.

Uso de tesauros

Mientras que el stemming persigue unir variantes morfológicas de palabras, el objetivo de usar un tesauro en el contexto de la reducción de dimensionalidad es unir sinónimos, es decir, palabras con significado similar. Generalmente, un tesauro es una colección de palabras que se agrupan en función de semejanzas en su significado. Por ejemplo, un tesauro puede contener relaciones entre palabras tales como "es similar a", "es más específico que", o "es más general que" para describir sinónimos, hipónimos, o hiperónimos, respectivamente. Estas relaciones se puede utilizar para agrupar palabras en clases semánticas equivalentes. La dimensión de los vectores de documentos se puede reducir por medio de una asociación de las características en la clase equivalente a la que pertenecen.

Obviamente, un tesauro es dependiente del idioma y del dominio de aplicación, construyéndose a menudo manualmente. Pero también nos encontramos con intentos de construir automáticamente tesauros basados en colecciones de documentos [Salton, 1989].

Agrupación de términos

Se trata de buscar relaciones de sinonimia entre palabras para agruparlas bajo una única representación, disminuyendo así el conjunto de características.

Se puede realizar una agrupación no supervisada buscando términos semejantes mediante alguna medida de similitud, o buscando co-ocurrencias o co-ausencias de los distintos términos en los documentos de entrenamiento. También se puede aplicar un agrupamiento supervisado,

aglutinando aquellos términos que tienden a estar presentes en la misma categoría o grupo de categorías.

En [Baker y McCallum, 1998] se presentan experimentos donde sólo se produce un 2 % de pérdida de la efectividad al agrupar conjuntos de 1000 términos e identificarlos utilizando una única característica. Sin embargo, los resultados obtenidos con métodos no supervisados son pobres [Lewis, 1992a; Li y Jain, 1998].

PCA, LSA y LDA

En una representación basada en palabras, podemos tener documentos que tratan del mismo tema pero que utilizan diferente vocabulario. Por ello, su representación vectorial puede ocultar su similitud temática. Si trabajamos con bolsas de palabras obtenidas directamente del conjunto de entrenamiento, sin ningún tipo de transformación, generamos características ignorando la relación que pueda existir entre los diferentes términos. Tal como hemos visto, podemos agrupar ciertas palabras utilizando un listado fijo de sinónimos. Pero existen relaciones adicionales que se pueden observar al analizar el corpus específico con el que se esté trabajando. Por ejemplo, si consideramos los conceptos que representa cada palabra, posiblemente detectemos redundancia de información, pues tal vez estemos utilizando características diferentes para representar conceptos muy similares. Por ejemplo, las palabras "naranja", "manzana", o incluso "zumo" guardan una estrecha relación, pues todas ellas se pueden identificar con el término o concepto "fruta".

Para conseguir aprovechar esta similitud en los términos disponemos de técnicas de extracción de características tales como PCA (*Principal Component Analysis* - Análisis de componentes principales), LSA (*Latent Semantic Analysis* - Análisis semántico latente) y LDA (*Linear Discriminant Analysis* - Análisis discriminante lineal). Son técnicas que se centran en capturar la temática de un documento para intentar solucionar el problema de la sinonimia, que aparece al utilizar directamente los términos que encontramos en los documentos del corpus.

PCA es un método estadístico que se basa en la idea de que cuanto más diferentes sean las características que representan un documento, más información representarán. Para ello, se eligen aquellas características cuyo peso tenga mayor diferencia respecto a la media. Aunque existen comparativas que justifican su uso [Lam y Lee, 1999; Busagala et al., 2005], debido a su alta complejidad, se ha utilizado sin embargo en pocas ocasiones dentro del ámbito de la clasificación de textos.

Mediante LSA [Schütze et al., 1995] se intenta capturar la temática (o la estructura semántica latente) de un documento analizando las co-ocurrencias entre términos. Se basa en el principio de que las palabras que aparecen en los mismos contextos tienen significados similares. Se utilizan técnicas de procesamiento del lenguaje natural para analizar relaciones entre los documentos que pertenecen a las mismas categorías y los términos que contienen para generar un número reducido de conceptos que puedan ser utilizados como características. Es decir, se pretende agrupar las palabras basándose en su aparición en documentos similares

y tomar como característica un concepto representativo de cada grupo formado.

LDA utiliza la misma idea que LSA, pero analiza además la distribución de los términos entre los documentos relevantes y no relevantes para cada categoría en particular. Sin embargo, a pesar de tener en cuenta más información respecto al contexto de co-ocurrencia, existen estudios que demuestran que no se obtienen mejores resultados respecto a LSA [Aggarwal y Zhai, 2012].

LSA y LDA resultan particularmente útiles cuando nos encontramos con un gran número de términos que contribuyen a representar una pequeña (aunque importante) cantidad de información. Consideremos por ejemplo el tema o categoría "Alternativas a las terapias tradicionales contra el cáncer", del que podemos encontrar gran cantidad de artículos, que contienen entre otros los siguientes términos: terapia genética, inmunización, terapia con vitamina A, transfusión de sangre del cordón umbilical, etc. Cada terapia tiene términos que son únicos, de forma que el conjunto final de términos relevantes es muy grande. LSA y LDA se pueden utilizar como método de compresión, pues pueden capturar la información relativa a las terapias alternativas en unas pocas características, eliminando la redundancia que acumularíamos si utilizamos cada una de las palabras.

3.2.3. Selección estadística

Como veremos más adelante, observando ciertos datos tales como la cantidad de apariciones de cada término, o el número de documentos que lo contienen, se pueden extraer diversas estadísticas para deducir la importancia de cada término dentro del corpus. Una vez establecida una ponderación, ésta nos permitirá ordenar por relevancia todas las características del corpus que son potencialmente válidas. Estableciendo un umbral de corte, de dicha lista se pueden descartar aquellos elementos estadísticamente menos importantes.

Entre dichas técnicas podemos nombrar la *ganancia de información*, *chi-cuadrado*, o la *tasa impar* entre otras. Las examinaremos en detalle dentro del contexto de ponderación de términos (sección 3.3.2).

3.2.4. Combinación de métodos

En esta última sección hemos expuesto diversas técnicas que permiten reducir el tamaño del conjunto del vocabulario, cuyos elementos corresponden con dimensiones de los vectores que representan los documentos. Y aunque cada una de las técnicas se puede aplicar individualmente, muchos autores utilizan varias combinaciones de las mismas. Por ejemplo, es muy común eliminar primero las palabras de parada basándose en una lista proporcionada manualmente. Después, las palabras restantes a menudo se reducen a sus raíces. Y a continuación, también resulta habitual eliminar los términos que aparezcan muy pocas veces, por ejemplo, aplicando un umbral a la frecuencia de aparición, o con la correspondiente frecuencia de documentos. Por último, se puede aplicar un método más elaborado, tal como la ganancia

de información,[4] para reducir aún más el número de términos.[5]

3.3. Ponderación de términos

Una vez hemos determinado el conjunto de características que representarán los textos, resulta conveniente aplicar una ponderación de acuerdo con su relativa importancia dentro de cada documento y de la colección de entrenamiento en general. Tal como vimos anteriormente (sección 2.3.3), esto resulta de gran relevancia para la construcción de clasificadores basados en vectores. De hecho, podemos encontrar numerosos estudios que demuestran que se puede mejorar considerablemente la calidad del clasificador utilizando técnicas específicas que permitan obtener los pesos adecuados [Debole y Sebastiani, 2004b; Liu y Loh, 2007; Lan et al., 2009].

Existen dos opciones consolidadas a tener en cuenta a la hora de representar los documentos. Podemos optar por la representación binaria, o por la representación ponderada, que es propia de casi todos los modelos modernos. La representación binaria implica que la única información que consideramos acerca de los términos es la de si están o no presentes en cada documento. Si eligiéramos esta opción básicamente estaríamos renunciando a la posibilidad de cualquier tipo de estimación, ya que la única valoración de las expresiones es que hayan superado algún umbral numérico establecido para reducir la dimensionalidad, o que simplemente hayan pasado el filtro de eliminación de palabras de parada.

La representación ponderada de los documentos consiste en un proceso que incluye procedimientos destinados a averiguar el valor representativo y discriminativo de un término en el contexto de un documento y en el contexto de la colección. Durante este proceso se asignan los pesos que cuantifican la importancia relativa de cada característica en función de su descripción de los contenidos de cada documento y de su capacidad para abstraer la temática a la que se encuentra asociado dicho documento. Se trata, por tanto, de ponderar la carga semántica y la capacidad descriptiva, lo que puede ayudarnos a establecer rankings y en general a producir representaciones vectoriales más matizadas.

Ya hemos mencionado la importancia de calcular el peso de las expresiones utilizando tres factores: local, global y normalización. Nos permitimos recordar aquí la fórmula a la que hacemos referencia:

$$w(t,d) = \frac{w_{local}(t,d) \cdot w_{global}(t)}{w_{norm}(d)} \quad . \tag{3.2}$$

Para intentar abordar mejor el contenido de los trabajos previos en este ámbito, dividiremos esta sección también en tres apartados, uno por cada uno de dichos factores. En el caso del factor local observaremos que se valoran las apariciones del término en el documento, mientras que para el factor global se intenta establecer la importancia de cada término en función

[4] Ver sección 3.3.2 para más detalles.

[5] Para más detalles sobre las combinaciones de diversas técnicas de reducción de dimensionalidad, se pueden consultar [Joachims, 1997; Dumais et al., 1998; Nigam y Ghani, 2000].

principalmente de sus apariciones en las distintas categorías de la colección de entrenamiento. Por último veremos que la normalización se puede utilizar para compensar las discrepancias en las longitudes de los documentos.

Resumiendo, en esta etapa de representación de documentos ya contamos con una serie de características válidas y lo que queda, por tanto, es aplicar una fórmula con la que ponderarlas para proceder finalmente a la generación de vectores. En esta sección presentaremos algunas de dichas fórmulas (partiendo del modelo básico expuesto en la ecuación 3.2), que nos ofrecerán diversas formas de asignar un valor determinado a cada uno de los términos extraídos de los documentos (en [Lan et al., 2009] podemos encontrar una recopilación bastante completa de tales métodos). Al presentar distintas posibilidades podremos evaluar las ventajas e inconvenientes de cada una, pues no existe una elección óptima que balancee perfectamente la precisión en los resultados y la complejidad computacional requerida para calcular dichos valores.

3.3.1. Factor local

Habiendo eliminado las palabras de parada, asumimos que la importancia de un término se incrementa con el número de apariciones. Por ello, una de las mejores formas de ponderar un término dentro de un texto concreto es contabilizar el número de veces que dicho término aparece [Salton et al., 1988]. Para obtener un peso local bastará con utilizar alguna función que devuelva un valor proporcional al número de apariciones, de forma similar a como se hace en la fase de reducción de dimensionalidad, considerando lo que se conoce como frecuencia de términos (*Term Frequency - TF*). Así, algunos de los factores locales utilizados habitualmente son:

Universitat d'Alacant
Universidad de Alicante

- **TF**. Número de apariciones del término en el documento:

$$w_{local}(t,d) = TF(t,d) \ , \tag{3.3}$$

  donde $TF(t,d)$ es la frecuencia del término t en el documento d.

- **Logaritmo de TF**. Se calcula de la siguiente forma:

$$w_{local}(t,d) = 1 + log \ TF(t,d) \ . \tag{3.4}$$

  Esta función fue propuesta en [Buckley, 1993]. Viene motivada por el hecho de que si por ejemplo un documento sólo tiene un término relevante con una frecuencia de aparición muy alta, entonces es probable que dicho documento se considere más importante que otro que tenga dos términos relevantes con frecuencias algo menores. Por lo tanto, en general se debería intentar que pocas ocurrencias de términos muy frecuentes no resulten en una valoración mayor que muchas ocurrencias de términos algo menos frecuentes.

La solución matemática para lograr esto es suavizar el valor en cuestión mediante un logaritmo.

Consideremos por ejemplo la categoría "reciclado de neumáticos" y dos documentos asociados a ella: D_1, con ocho apariciones de la palabra "reciclar" y D_2, que utiliza las palabras "reciclado" y "neumáticos" tres veces cada una. Si todo lo demás se plantea en las mismas condiciones, entonces empleando la simple frecuencia de términos, D_1 adquirirá una mejor valoración (proporcional a 8) que D_2 (que tendrá una similitud proporcional a $3 + 3 = 6$). Pero intuitivamente, se puede deducir que D_2 refleja mejor el contenido de la categoría porque menciona todos los conceptos. Cuando se utiliza el logaritmo, la relevancia de D_1 viene dada por $1 + log(8) = 1,9$, mientras que D_2 adquiere una valoración de $[1 + log(3)] + [1 + log(3)] = 2,95$, de forma que D_2 será finalmente mejor considerado que D_1, tal como debería ser.

- **Valor binario**. Se descarta por completo el número de apariciones de un término. Simplemente se considera un valor de uno o cero dependiendo de la presencia (1) o ausencia (0) del término en el texto. Aunque su utilidad real es muy reducida, este factor proporciona una medida muy simple que se puede emplear como base para comparar las mejoras resultantes al utilizar otros métodos de ponderación.

- **TF aumentada**. Se pretende reducir el rango de las posibles contribuciones de la frecuencia de un término. Esto se puede conseguir comprimiendo el rango de valores de TF (por ejemplo entre $0,5$ y 1). El objetivo es conseguir que la mera presencia de un término en un texto tenga algún valor por defecto ($0,5$ por ejemplo). Apariciones adicionales de un término podrían incrementar el peso del mismo hasta algún valor máximo (habitualmente 1). Esto se puede formalizar mediante la siguiente ecuación:

$$w_{local}(t,d) = 0,5 + 0,5\frac{TF(t,d)}{max\ TF}\ . \tag{3.5}$$

El valor TF aumentado también enfatiza la presencia de muchos términos relevantes frente a la de unos pocos, al igual que ocurre con su equivalente logarítmico. Una simple aparición de un término al menos contribuye con un valor de $0,5$ y frecuencias altas sólo pueden incrementar dicho valor hasta 1. De esta forma se consiguen evitar resultados erróneos causados por grandes variaciones en la longitud de los documentos de entrenamiento.

Debido a la baja complejidad y alta efectividad que el método de ponderación mediante frecuencia de términos ofrece en situaciones reales, hay pocos estudios que proponen mejoras. Sin embargo, presenta el inconveniente de asignar el peso unitario a gran cantidad de expresiones. Tengamos en cuenta que al tratarse de un peso local, estamos considerando las repeticiones de palabras en un solo documento, con lo que es muy probable que nos encontremos con expresiones que sólo aparecen una vez y que consecuentemente reciben el mismo peso. Basándose en este hecho, en [Hassan y Banea, 2006] se presenta una alternativa de

ponderación local denominada *random walk*, que utiliza grafos para representar el documento y valorar las palabras según el contexto donde se encuentra cada una [Mihalcea y Tarau, 2004]. Se consiguieron mejoras importantes para una gran variedad de algoritmos de clasificación, reduciendo los errores en los resultados entre 3,5 % y 44 %, comparando con el modelo básico de TF. Sin embargo, no podemos dejar de lado la sofisticación del método empleado, que dista mucho del uso de una simple estadística.

3.3.2. Factor global

Utilizar simplemente el factor local para estimar la importancia de un término no resulta suficiente. Consideremos por ejemplo las palabras comunes que aparecen con una frecuencia muy alta en numerosos documentos. Algunas de ellas no son informativas en absoluto y ya se eliminaron en la fase de reducción de dimensionalidad. Para el resto podemos aplicar la misma filosofía, planteando una valoración de las expresiones mediante algún método global, que observe las apariciones de cada palabra en toda la colección de entrenamiento para intentar deducir si su uso es muy común y por lo tanto sopesar su capacidad de predicción de una categoría en particular. Más concretamente, los términos mejor valorados deberían compartir alguna de las siguientes propiedades [Lertnattee y Theeramunkong, 2004]:

- Deberían aparecer frecuentemente en una clase específica.
- Deberían aparecer en pocos documentos.
- Deberían estar distribuidos de manera muy diferente entre las distintas clases.
- No deberían estar distribuidos de manera muy diferente entre los documentos de la misma clase.

Universitat d'Alacant
Universidad de Alicante

Para intentar reflejar estas consideraciones y especificar formalmente algunas de las propuestas existentes para el cálculo de pesos globales utilizaremos la siguiente notación:

- N = Número total de documentos en el conjunto de aprendizaje.
- $P(c_i|t_j)$ = Número de documentos pertenecientes a la categoría c_i que contienen el término t_j.
- $P(\overline{c_i}|t_j)$ = Número de documentos que no pertenecen a la categoría c_i y que contienen que el término t_j.
- $P(c_i|\overline{t_j})$ = Número de documentos pertenecientes a la categoría c_i que no contienen el término t_j.
- $P(\overline{c_i}|\overline{t_j})$ = Número de documentos que no pertenecen a la categoría c_i y que no contienen el término t_j.

- $P(c_i)$ = Número de documentos pertenecientes a la categoría c_i.
- $P(t_j)$ = Número de documentos que contienen el término t_j.
- $P(\overline{c_i})$ = Número de documentos que no pertenecen a la categoría c_i.
- $P(\overline{t_j})$ = Número de documentos que no contienen el término t_j.

Como se observa, todos estos valores se pueden obtener fácilmente contabilizando el número total de documentos que pertenecen a cada categoría o aquellos que contienen (o no contienen) un término específico.

También hay que tener en cuenta que las ecuaciones que se plantean a continuación proporcionan el peso específico de un término para una categoría en particular ($w(t_j, c_i)$). Sin embargo, algunos de los clasificadores que detallaremos más adelante (sección 3.4) necesitan un único valor que represente el peso de cada término con respecto a todo el corpus. Para ello deberemos globalizar los cálculos mediante alguna técnica que compendie los valores de todas las clases en uno solo ($w_{global}(t_j)$). Esto se puede conseguir mediante alguno de los siguientes cálculos [Debole y Sebastiani, 2003]:

- Suma:

$$w_{global}(t_j) = \sum_{i=1}^{C} w(t_j, c_i) \ , \tag{3.6}$$

- Suma ponderada:

$$w_{global}(t_j) = \sum_{i=1}^{C} \frac{P(c_i)}{N} \cdot w(t_j, c_i) \ , \tag{3.7}$$

- Valor máximo:

$$w_{global}(t_j) = \max_{i=1}^{C} w(t_j, c_i) \ , \tag{3.8}$$

donde C es el número total de categorías del corpus y $w(t_j, c_i)$ puede venir dado por cualquiera de los métodos de ponderación que se expondrán a continuación: inversa de la frecuencia del documento, Chi-cuadrado, ganancia de información, tasa impar, intervalo de confianza y frecuencia de relevancia.

Inversa de la frecuencia del documento (IDF)

Se ha observado que las palabras que aparecen en muchos documentos resultan menos útiles que aquellas que se emplean sólo en unos pocos textos [Sparck-Jones, 1973]. Esto es debido a que la capacidad discriminativa de una palabra disminuye cuando aumenta el número de documentos en los que aparece, ya que las expresiones muy comunes no nos permiten caracterizar temas específicos. Basándonos en esta afirmación podemos utilizar la

inversa de la frecuencia del documento (*Inverse Document Frequency - IDF*) para valorar la importancia de un término t_j dentro de un conjunto de documentos. En particular para la categoría c_i será:

$$IDF(t_j, c_i) = log \ \frac{P(c_i)}{P(c_i|t_j)} \ . \tag{3.9}$$

Así pues, una expresión que aparece en todos los documentos de una colección obtendrá un peso nulo.

Respecto a su uso, raramente se utiliza IDF como medida de selección de características, pues se consiguen resultados muy similares empleando listas fijas de expresiones muy frecuentes (stopwords). Sin embargo, su uso sí es muy frecuente en la ponderación de expresiones, combinada con el factor local TF, pues se ha demostrado su equivalencia (o incluso superioridad en algunos casos) respecto a medidas más sofisticadas con una complejidad muy superior. Aún así, encontramos algunos intentos por mejorar los resultados obtenidos con IDF, entre los cuales podemos nombrar χ^2, IG, OR, ConfWeight y RF, que se expondrán a continuación. Pero hay que tener en cuenta que tanto TF como IDF se pueden obtener simplemente contabilizando las apariciones de las palabras en la colección de documentos, pudiéndose actualizar fácilmente la ponderación si se añaden nuevos textos de entrenamiento. Las propuestas que describimos a continuación están basadas en la observación de la distribución de las características entre las muestras positivas y negativas de cada categoría y, por lo tanto, llevan implícito un incremento en la complejidad necesaria para su cálculo.

Universitat d'Alacant
Universidad de Alicante

Chi-cuadrado (χ^2)

Su aplicación se basa en la intuición de que un término es útil para discriminar entre diversas clases si su frecuencia de aparición depende principalmente de la categoría en la que dicho término aparece.

La medida Chi-cuadrado (*Chi-Square* - χ^2) se emplea en estadística para evaluar la independencia de dos eventos. Si la utilizamos para valorar una determinada expresión, los dos eventos corresponderán con la ocurrencia de dicha expresión (t_j) y la ocurrencia de una determinada categoría (c_i). Da una medida de cuánto se desvía la frecuencia esperada de la frecuencia observada. Es decir, si se cumple la hipótesis de que el término t_j es independiente de la categoría c_i, el valor de χ^2 debería ser nulo. Esto indicaría que el término no aparece en ningún documento de entrenamiento de la categoría en cuestión. Por otro lado, un valor grande de χ^2 debe significar que la hipótesis de independencia es incorrecta, lo que implica que los valores observados y esperados difieren bastante y por lo tanto, c_i depende en gran medida de t_j. Dicha consideración se puede plasmar de la siguiente forma [Zheng et al., 2004]:

$$\chi^2(t_j, c_i) = \frac{N \cdot [P(c_i|t_j) \cdot P(\overline{c_i}|\overline{t_j}) - P(\overline{c_i}|t_j) \cdot P(c_i|\overline{t_j})]^2}{P(c_i) \cdot P(\overline{c_i}) \cdot P(t_j) \cdot P(\overline{t_j})} \ . \tag{3.10}$$

Desde un punto de vista estadístico, este método puede resultar problemático: cuando un test estadístico se emplea muchas veces, la probabilidad de obtener al menos un error se incrementa. Por ejemplo, si se rechazan 1.000 hipótesis de independencia, cada una de ellas con una probabilidad de error de $0,05$, entonces una media de $0,05 \cdot 1.000 = 50$ valoraciones serán incorrectas [Manning et al., 2008]. Este inconveniente es casi imperceptible en la ponderación de términos cuya frecuencia de aparición sea muy alta, pues el comportamiento de éstos es bastante predecible y, por lo tanto, la probabilidad de cometer errores es menor. Sin embargo, no resulta una medida muy fiable para términos con una frecuencia de aparición muy baja. Esto se confirma en [Dunning, 1993], donde se realizó un estudio con 2.693 bigramas. De éstos sólo 11 se consideraron adecuados para su valoración con la estadística χ^2 y todos ellos contenían la palabra the, que es la más usada en inglés. Debido a esto, la medida χ^2 se suele utilizar sólo en la fase de selección de características para filtrar las expresiones muy comunes. Su validez como función de ponderación es más reducida, pues en éste último caso las expresiones menos habituales son precisamente aquellas que conviene ponderar mejor.

También nos encontramos con el problema de que χ^2 presenta una complejidad temporal cuadrática [Yang y Pedersen, 1997], lo que se vuelve en su contra, puesto que disponemos de alternativas más sencillas que presentan un comportamiento similar. En [Debole y Sebastiani, 2003; Zhi-Hong Deng y Xie, 2004] encontramos experimentos que presentan resultados obtenidos utilizando pesos calculados mediante IDF y χ^2. En algunos casos se obtiene una precisión parecida con ambas medidas (se aprecian diferencias de poco más de 1 %), lo que no invita a utilizar χ^2 para ponderar características en colecciones de documentos de gran tamaño.

En [Yang y Pedersen, 1997] se comparan diversas técnicas de selección se características y en este caso se logran mejoras más apreciables respecto a otros métodos más básicos. Reduciendo el número de términos hasta un 90 %, χ^2 tiene un comportamiento equiparable a DF (ver sección 3.2.1), pero no es el caso para reducciones más agresivas, donde χ^2 ofrece mejor precisión en los resultados. Aunque como hemos comentado, el coste computacional se ve incrementando, pues DF presenta una complejidad lineal, mucho menor que la de χ^2.

Universidad de Alicante

Ganancia de información (IG)

Se basa en la asunción de que los términos más valiosos para la clasificación en una categoría en particular, son aquellos que se distribuyen de forma más desigual entre los conjuntos de muestras positivas y negativas de la categoría.

Más concretamente, la ganancia de información (*Information Gain - IG*) mide la cantidad de información que se puede utilizar para la predicción de una clase en particular conociendo la presencia o la ausencia de cada término dentro de los documentos de entrenamiento pertenecientes a dicha clase. Para obtener esta medida podemos utilizar la ecuación propuesta en [Lan et al., 2009]:

$$IG(t_j, c_i) = \frac{P(c_i|t_j)}{N} \cdot \log \frac{P(c_i|t_j) \cdot N}{P(c_i) \cdot P(t_j)} + \frac{P(c_i|\overline{t_j})}{N} \cdot \log \frac{P(c_i|\overline{t_j}) \cdot N}{P(c_i) \cdot P(\overline{t_j})} + \frac{P(\overline{c_i}|t_j)}{N} \cdot \log \frac{P(\overline{c_i}|t_j) \cdot N}{P(\overline{c_i}) \cdot P(t_j)} + \frac{P(\overline{c_i}|\overline{t_j})}{N} \cdot \log \frac{P(\overline{c_i}|\overline{t_j}) \cdot N}{P(\overline{c_i}) \cdot P(\overline{t_j})} \quad . \tag{3.11}$$

Si la distribución de un término es la misma en c_i que en toda la colección de entrenamiento, entonces IG es cero. Por otro lado, IG alcanzará su valor máximo cuando el término sea un indicador perfecto de pertenencia a una categoría, esto es, cuando el término se encuentre en un documento si y sólo si el documento está dentro del conjunto de entrenamiento de la categoría.

Se utilizó por primera vez en [Lewis y Ringuette, 1994] para la construcción de clasificadores basados en árboles de decisión (sección 2.3.4). En la actualidad también se conoce como información mutua media [Li-zhen et al., 2005]. Puesto que presenta una complejidad cuadrática (similar a χ^2) y en general no consigue una mejoría considerable cuando se emplea para ponderar características [Debole y Sebastiani, 2003], no se encuentran demasiados trabajos que demuestren su aplicación en este ámbito. Aunque en [Lertnattee y Theeramunkong, 2004] podemos observar un interesante estudio, éste es un caso aislado y bastante particular. Se obtuvieron resultados prometedores, pero se utilizaron dos factores globales ($IDF \cdot IG$) y los pesos propuestos sólo se emplearon para mejorar el funcionamiento de clasificadores basados en prototipos, intentando encontrar un centroide representativo de cada categoría. Además, las mejoras importantes se observaron al utilizar una colección de documentos del ámbito médico.

Sin embargo, IG sí ha sido utilizado en gran cantidad de ocasiones para seleccionar las características menos discriminativas (aquellas con el valor IG más bajo) y eliminarlas del vocabulario para reducir la dimensionalidad (ver por ejemplo [Benkhalifa et al., 1999; Forman, 2002; Debole y Sebastiani, 2004b; Li-zhen et al., 2005]). En este aspecto tenemos que destacar un serie de experimentos muy completos expuestos en [Forman, 2003], donde el uso de los términos seleccionados con IG ofrece la mejor precisión en la clasificación, realizando comparaciones contra otros 11 métodos de selección.

Tasa impar (OR)

Fue propuesta originalmente en [Van Rijsbergen, 1979] bajo el nombre de *Odds Ratio (OR)*. De forma similar a IG, se apoya en la idea básica de que la distribución de las características en los documentos relevantes es diferente de la distribución de las características en los documentos no relevantes. Se define de la siguiente forma [Zheng y Srihari, 2003]:

$$OR(t_j, c_i) = \frac{[P(c_i|t_j) + 0,5] \cdot [P(\overline{c_i}|\overline{t_j}) + 0,5]}{[P(c_i|\overline{t_j}) + 0,5] \cdot [P(\overline{c_i}|t_j) + 0,5]} \quad . \tag{3.12}$$

Los valores mayores que 1 corresponden a características indicativas de pertenencia, mientras que los valores menores que 1 corresponden a características indicativas de no pertenencia. De esta forma los valores más grandes se asignarán a aquellos términos que aparezcan frecuentemente en los documentos de entrenamiento de la categoría en cuestión (muestras positivas) y raramente en el resto de documentos (muestras negativas). Además, mediante la ecuación propuesta, aunque una característica aparezca muy pocas veces en las muestras positivas, conseguirá una ponderación alta si no aparece en ninguna muestra negativa.

Se usó con éxito en [Mladenić, 1998] para seleccionar términos y reducir la dimensionalidad. Con este propósito, los mismos autores la emplearon posteriormente en el ámbito de la categorización de páginas web, para construir perfiles de usuario [Mladenić, 2003] y también para tareas de clasificación utilizando las categorías propuestas por Yahoo! [Mladenić y Grobelnik, 2003]. En este último caso se llegaron a seleccionar sólo un 5 % de las características originales, pues además se emplearon otras técnicas de reducción tales como stopwords, eliminación de expresiones con una frecuencia muy baja y además, se utilizó una versión reducida de cada documento, en vez de emplear el texto completo. Los estudios mencionados utilizaron datos relativamente peculiares (basados en contenidos reales extraídos de Internet), pero también podemos encontrar otros autores que realizaron experimentos más comparables que corroboran el uso de OR ([Forman, 2003] en selección de términos y [Pekar et al., 2004] en ponderación de características). Aunque estos últimos resultados no fueron tan destacables, sí podemos encontrar una publicación relativamente reciente [Forman, 2008] que muestra una extensiva comparación de 30 métodos diferentes de ponderación, donde OR se sitúa en segundo lugar. Además, en la actualidad podemos encontrar algunas aplicaciones más específicas, como es el caso propuesto por [Ru et al., 2008], donde OR se utilizó con éxito para representar vectores de documentos en sistemas expertos.

Universitat d'Alacant
Universidad de Alicante

Intervalo de confianza (ConfWeight)

En [Soucy y Mineau, 2005] se presenta un método que intenta estimar el valor de cada término para ver si se encuentran dentro de un cierto intervalo de confianza y se debe asignar por lo tanto un peso superior a cero. Además considera la proporción de documentos entre las muestras positivas y negativas que contienen dicho término y, si la diferencia entre uno y otro conjunto es grande, el peso también será alto. Más concretamente, se considera como límite inferior el término que aparece menos veces en las muestras positivas ($MinPos$) y como límite superior el término que aparece mayor número de veces en las muestras negativas ($MaxNeg$):

$$ConfW(t_j, c_i) = \begin{cases} log_2 \left[\dfrac{2 \cdot MinPos(c_i)}{MinPos(c_i) + MaxNeg(c_i)} \right] & \text{si } MinPos(c_i) > MaxNeg(c_i) \\ 0 & \text{en otro caso} \end{cases} , \tag{3.13}$$

donde $MinPos(c_i)$ es el mínimo de todos los valores que puede tomar $P(c_i|t_j)$ y $MaxNeg(c_i)$ es el máximo de todos los posibles valores de $P(\overline{c_i}|t_j)$:

$$MinPos(c_i) = \min_{j=1}^{N} \frac{P(c_i|t_j) + 1,96}{N + 3,84} \;\; ; \;\; MaxNeg(c_i) = \max_{j=1}^{N} \frac{P(\overline{c_i}|t_j) + 1,96}{N + 3,84} \;\; . \qquad (3.14)$$

Tal como se demuestra en [Soucy y Mineau, 2005; Xu y Li, 2007; Coterillo, 2010], al representar los documentos mediante pesos obtenidos con ConfWeight se mejora sustancialmente la precisión de la clasificación con respecto a IDF, χ^2 e IG. Además, presenta una efectividad muy estable frente a incrementos en el número de características. Sin embargo, se observa un ligero inconveniente al incrementar el número de categorías, pues muestra un deterioro brusco en los resultados, siendo éste un comportamiento que no se presenta al utilizar IDF. La justificación la encontramos en el hecho de que IDF presenta la ventaja de que no emplea información particular relativa a las muestras de entrenamiento pertenecientes a cada categoría. De esta forma, la calidad de los pesos no se deteriora al incrementar el abanico de las posibles etiquetas que se puedan asignar a cada documento.

Frecuencia de relevancia (RF)

En último lugar presentamos la frecuencia de relevancia (*Relevance Frequency - RF* [Lan et al., 2006]). Esta medida sólo tiene en cuenta la consideración de que los términos más discriminativos serán aquellos que tengan mayor número de apariciones en las muestras positivas y menor número de apariciones en las negativas.

Utilizando la medida IDF nos podemos encontrar con expresiones que aparecen en un número distinto de documentos en el conjunto de muestras positivas y negativas, pero que sin embargo pueden recibir el mismo valor IDF si la suma de ambas cantidades es la misma. Es decir, en caso de que $P(c_i|t_j) + P(\overline{c_i}|tj) = P(c_i|t_k) + P(\overline{c_i}|t_k)$ también se cumplirá que $IDF(c_i, t_j) = IDF(c_i, t_k)$, aunque el número de muestras positivas sea distinto para cada término: $P(c_i|t_j) \neq P(c_i|t_k)$. RF pretende evitar estas situaciones mediante una de las formas más sencillas posibles, calculando simplemente la proporción entre las muestras positivas y negativas:

$$RF(t_j, c_i) = log_2 \left[2 + \frac{P(c_i|t_j)}{max(1, P(\overline{c_i}|tj))} \right] \;\; . \qquad (3.15)$$

Se puede observar que el valor mínimo se obtiene en el caso de que el término no aparezca entre las muestras positivas. En el extremo opuesto, el valor obtenido será máximo si y sólo si el término aparece en todos los documentos representativos de la clase. Comparado con otras medidas, también se puede apreciar que no se utiliza el valor $P(\overline{c_i}|\overline{t_j})$. Esto se justifica en el hecho de que este último valor es más grande que cualquiera de las otras frecuencias de aparición y, por lo tanto, su uso puede ocultar las variaciones en el resto de frecuencias. De hecho, el nombre se eligió basándose en esta consideración, pues para el cálculo propuesto sólo se tienen en cuenta los documentos *relevantes*, es decir, aquellos que contienen el término.

Pese a la simplicidad de su planteamiento, según sus autores ofrece los mejores resultados publicados hasta la fecha respecto a la ponderación de características supervisada, es decir, utilizando la información de pertenencia o no pertenencia de cada documento a las distintas categorías. En [Lan et al., 2009] podemos consultar un exhaustivo estudio que así lo demuestra al compararlo con gran variedad de métodos, pasando por IDF, χ^2 e IG. Desafortunadamente, los resultados presentados no han sido confirmados todavía en publicaciones de otras fuentes, pues casi todos los estudios existentes han sido realizados por los mismos autores, tal como se observa en la bibliografía [Man Lan, 2005; Lan et al., 2005, 2006, 2009]. Aparte de estas citas, sólo nos encontramos con [Liu et al., 2007], donde se evalúa la calidad del proceso de clasificación generando variaciones en los pesos de los términos de documentos de entrenamiento pertenencientes a clases muy descompensadas. En este caso, χ^2 e IG resultaron ser superiores en una evaluación media, pero inferiores a RF en la evaluación individual de algunas categorías.

3.3.3. Factor de normalización

El tercer y último factor de ponderación consiste en la normalización de los vectores de documentos, que ha resultado de probada utilidad especialmente en colecciones con documentos de tamaño muy diferente [Singhal et al., 1996; Zu et al., 2003]. Habitualmente, los documentos cortos tienden a estar representados por vectores de longitud reducida, mientras que los documentos de mayor extensión tienden a generar mayor número de características significativas [Salton et al., 1988]. Cuando se usa un número mayor de términos para representar un documento, aumentamos la posibilidad de encontrar similitudes entre dicho documento y otros que puedan presentarse. Este hecho provocaría que las clases con mayor probabilidad de salir elegidas en el proceso de clasificación fueran aquellas relacionadas con los documentos con mayor contenido textual (una decisión errónea en muchos casos), lo que sugiere el uso de un componente de normalización que se aplique sobre cada uno de los pesos para intentar igualar la longitud de los vectores.

Alguna de las propuestas existentes se expone a continuación. Considerando V como el número de elementos en el conjunto del vocabulario de términos que componen un documento d, se puede calcular $w_{norm}(d)$ aplicando alguna de estas aproximaciones:[6]

- **Normalización mediante la suma de todos los pesos**. En los documentos grandes, el peso local de cada una de sus características será probablemente alto, debido al incremento en las frecuencias de aparición (TF) de cada término. Además, también tendrá mayor número de términos, lo que ocasionará un incremento en el valor del documento, causado por las contribuciones de las frecuencias de dichos términos. Al dividir los pesos por el valor generado mediante esta propuesta penalizaremos tales incrementos, pues tanto el uso de términos con un alto peso, como el uso de gran

[6] Conviene recordar que el factor de normalización corresponde con el denominador de la función de ponderación propuesta en la fórmula 3.2.

cantidad de términos, se reflejará en un factor de normalización alto:

$$w_{norm}(d) = \sum_{i=1}^{V}[w_{local}(t_i, d) \cdot w_{global}(t_i)] \ . \tag{3.16}$$

- **Normalización del coseno**. Además de tener en cuenta las consideraciones expuestas en la simple suma, este método pretende anticiparse a futuros cálculos. Como veremos más adelante (sección 3.4.6), una de las formas de evaluar la similitud entre documentos es utilizando el cálculo del coseno entre sus vectores (por ejemplo, para intentar deducir clases de pertenencia observando la proximidad de un nuevo documento con respecto a los que ya se encuentran etiquetados). Para ello, los vectores de los documentos implicados deben tener longitud unitaria. Esto se consigue dividiendo cada uno de los pesos entre la longitud total del vector, que se calcula de la siguiente forma:

$$w_{norm}(d) = ||\vec{d}|| = \sqrt{\sum_{i=1}^{V}[w_{local}(t_i, d) \cdot w_{global}(t_i)]^2} \ . \tag{3.17}$$

  Mediante este método, al normalizar off-line los pesos de las representaciones vectoriales de los documentos de entrenamiento, optimizaremos los cálculos posteriores que se realicen on-line para evaluar por ejemplo similitudes con nuevos documentos.

3.4. Algoritmos de clasificación

Una gran variedad de algoritmos de aprendizaje y clasificación han sido estudiados y comparados en la literatura (ver por ejemplo [Yang y Liu, 1999]). A continuación describimos algunos de los que se han usado ampliamente para abordar el aprendizaje supervisado. Más concretamente, dentro del ámbito de investigación en el que vamos a trabajar, consideraremos principalmente tres tipos de clasificadores. Primero, describiremos aquellos que utilizan funciones de distancia o similitud entre prototipos, tales como los basados en Rocchio o en la regla de los vecinos más cercanos. En segundo lugar describiremos las máquinas de vectores de soporte, que intentan separar de forma óptima las muestras positivas de las negativas en cada categoría. En tercer lugar mencionaremos el clasificador de Naive Bayes, correspondiente al tipo de clasificadores probabilísticos. Por último, haremos hincapié en las aproximaciones que intentan combinar los resultados obtenidos por distintos clasificadores para ofrecer una sola respuesta que sea más precisa que cualquiera de las obtenidas por un solo algoritmo de forma individual.

3.4.1. Vecinos más cercanos (kNN)

La idea básica, tal como se plantea en [Yang y Liu, 1999], es que dado un documento de prueba, se puede diseñar un sistema que encuentre los vecinos más cercanos entre los

documentos de entrenamiento con el propósito de utilizar las categorías a las que pertenecen los vecinos encontrados para proponer unas categorías candidatas al documento de prueba.

En su forma más simple, este algoritmo calcula la similitud entre el documento que se quiere clasificar y cada uno de los documentos de entrenamiento ofreciendo como resultado las categorías del documento más cercano. Sin embargo, existen algunas variantes más conocidas, como por ejemplo, la de los k-ésimos vecinos más cercanos, que consiste en utilizar los k documentos más parecidos en lugar de sólo el primero. En este caso, para cada uno de los k documentos elegidos, se mide el nivel de similitud con respecto al nuevo documento y se utiliza dicho valor como peso de las categorías del documento vecino. Si varios de los k-ésimos vecinos comparten una categoría, entonces se suman los pesos de cada uno de los vecinos y la suma resultante se usa como la probabilidad de esa categoría con respecto al documento de prueba. Finalmente, se genera una lista ordenada de las categorías candidatas. En [Yang y Liu, 1999] se formaliza la regla de decisión así:

$$y(x, c_j) = \sum_{i=1}^{k} sim(x, d_i) \cdot y(d_i, c_j) \ , \tag{3.18}$$

donde:

- $y(x, c_j)$ es la valoración de pertenencia del nuevo documento x respecto a la categoría c_j .
- $y(d_i, c_j)$ es la clasificación para el documento d_i con respecto a c_j, es decir es $y = 1$ si d_i pertenece a la categoría c_j y es $y = 0$ en caso contrario.
- $sim(x, d_i)$ es la similitud entre el documento de prueba x y el documento de entrenamiento $\vec{d_i}$.

Para llevar a la práctica este concepto no se requiere ningún proceso de aprendizaje. Sólo es necesario almacenar los vectores correspondientes a todos los documentos de entrenamiento y comparar el vector que representa al nuevo documento con todos los almacenados para decidir cuáles de ellos son los k-vecinos más cercanos y así extraer las clases de pertenencia. Por ello se dice que este tipo de algoritmos son perezosos, porque no evalúan los documentos clasificados por los expertos hasta un momento antes de emitir su decisión de clasificación.

Resumiendo, el proceso de clasificación basado en los k-vecinos más cercanos, se puede resolver fácilmente en 4 pasos:

1. Se compara el nuevo documento con cada uno de los documentos preclasificados. Como resultado de esta comparación se elabora un ranking en el que los documentos preclasificados más similares al documento que queremos clasificar aparecen más arriba.

2. Se seleccionan los k documentos preclasificados más similares al documento a clasificar. K es un valor establecido de forma aleatoria, pero se suele proponer un valor que oscila entre 2 y 10.

3. Se considera la similitud de cada documento preclasificado con respecto del nuevo documento como un voto a favor de la categoría a la que pertenece el documento preclasificado.

4. Se selecciona la categoría más votada y se clasifica el nuevo documento bajo la misma. En realidad podemos considerar los valores de la decisión de clasificación sumándolos, en lugar de considerarlos como un voto, lo que mejora los resultados, aunque sólo es posible si el sistema es capaz de crear estimaciones graduales de la similitud de un documento con respecto de otro. Por ejemplo, si hemos utilizado una representación de los documentos binaria no podremos aprovecharnos de estas características.

A diferencia de otros algoritmos, una de las ventajas de los clasificadores basados en la regla de los k-vecinos más cercanos es la carencia de un proceso específico de aprendizaje. Pero como contrapartida, cada vez que el algoritmo recibe un nuevo documento, éste se debe comparar con cada una de las muestras de entrenamiento para poder emitir una decisión. De esta forma, la sencillez conseguida en el diseño del clasificador se vuelve contra nosotros. Debido al alto número de comparaciones que se proponen, nos encontramos con un coste computacional que hace que este algoritmo sea ineficiente si se pretende trabajar con corpus de gran tamaño.

Universitat d'Alacant
Universidad de Alicante

3.4.2. Rocchio

Se basa en un concepto con varias décadas de antigüedad [Rocchio, 1971], que ha sido ampliamente estudiado, adaptado, modificado y perfeccionado en numerosas ocasiones (ver por ejemplo [Chen et al., 2002]). La idea es muy simple: si conseguimos representar cada categoría mediante un solo prototipo[7] (o vector representativo), el proceso de clasificación se simplifica bastante, pues bastará con evaluar la similitud del nuevo documento con cada uno de los prototipos para obtener un ranking de categorías. De esta forma, puede resultar sencillo desarrollar un algoritmo que utilice este planteamiento, ya que por ejemplo, se puede obtener un prototipo que represente a todos los documentos de entrenamiento pertenecientes a una clase calculando simplemente la media entre todos ellos.

Para construir un clasificador binario mediante esta técnica, sólo será necesario obtener dos representaciones por cada categoría: una para el conjunto de las muestras positivas y otro para el de las negativas. En este caso, para decidir si un nuevo documento pertenece o no a una determinada clase, sólo debemos calcular su distancia a los dos prototipos representativos y ofrecer la decisión de pertenencia en función de su cercanía a uno u otro.

[7] Gran parte del trabajo de esta tesis se basa en esta filosofía.

En otras palabras, una vez se tienen los prototipos de cada una de las clases, el proceso de entrenamiento o aprendizaje está concluido. Para categorizar nuevos documentos, simplemente se estima la similitud entre el nuevo documento y cada uno de los prototipos. El que devuelve un índice mayor nos indica la categoría que se debe asignar a ese documento.

Este tipo de clasificadores ofrecen muy buenos resultados con una complejidad computacional muy baja y, por ello, el algoritmo de Rocchio ha sido utilizado frecuentemente en tareas de categorización [Lewis et al., 1996]. En [Joachims, 1997] se realizaron experimentos con varios grupos de noticias de Internet, así como con noticias de la agencia de prensa Reuters, con excelentes resultados. En [Figuerola et al., 2001] se obtuvieron resultados con noticias de prensa en español comparables a los obtenidos por clasificadores humanos.

A diferencia del método de los vecinos más cercanos, los algoritmos tipo Rocchio sí dependen de un proceso de aprendizaje, que aunque obligatorio, implica simplemente el promediado de los vectores de los documentos de entrenamiento. De hecho, ni siquiera se requiere de ningún proceso de optimización. Por ello, la complejidad temporal para completar la construcción del clasificador es proporcional al número de documentos en el conjunto de entrenamiento y el número de términos en el vocabulario. Sin embargo, el tiempo necesario para clasificar un nuevo documento crece linealmente sólo con el número de categorías y la cantidad de términos. Este valor no se verá afectado aunque incrementemos el número de muestras utilizadas para el aprendizaje, lo que nos llevará habitualmente a un tiempo de respuesta mucho menor que el ofrecido por kNN.

3.4.3. Máquinas de vectores de soporte (SVM)

Este método fue propuesto por primera vez en [Joachims, 1998] para su uso en clasificadores de texto. El problema a resolver se puede entender fácilmente en términos geométricos (ver figuras 3.2 y 3.3), imaginando una superficie en el espacio vectorial que consiga separar en dos partes el conjunto de las muestras de entrenamiento, creando dos subconjuntos, uno que contenga las muestras positivas y otro que contenga las negativas.

Intentaremos ampliar mejor el concepto partiendo de la figura 3.3. Por simplicidad, se muestra un caso donde los vectores se encuentran distribuidos en un espacio de dos dimensiones y que, además, pueden ser separados linealmente. Se genera un hiperplano (o superficie de decisión), que es la línea que separa las muestras positivas de las negativas en el espacio vectorial. Las líneas continuas muestran dos posibles hiperplanos, cada uno de los cuales separa correctamente los dos grupos de datos. Las líneas discontinuas paralelas a las continuas muestran el margen en el que se puede mover la superficie de decisión sin cometer errores en la clasificación. El problema que SVM pretende resolver se basa en encontrar el hiperplano que maximiza el margen entre los puntos que representan el conjunto de entrenamiento. Más concretamente, durante el aprendizaje intentaremos encontrar el vector $\vec{w}$ (hiperplano) y la constante b (margen) que para cualquier documento $\vec{x}$ satisfagan la ecuación $\vec{w} \cdot \vec{x} - b > 0$ en caso de que el documento de prueba esté dentro del conjunto de pertenencia de la categoría y $\vec{w} \cdot \vec{x} - b < 0$ en caso contrario.

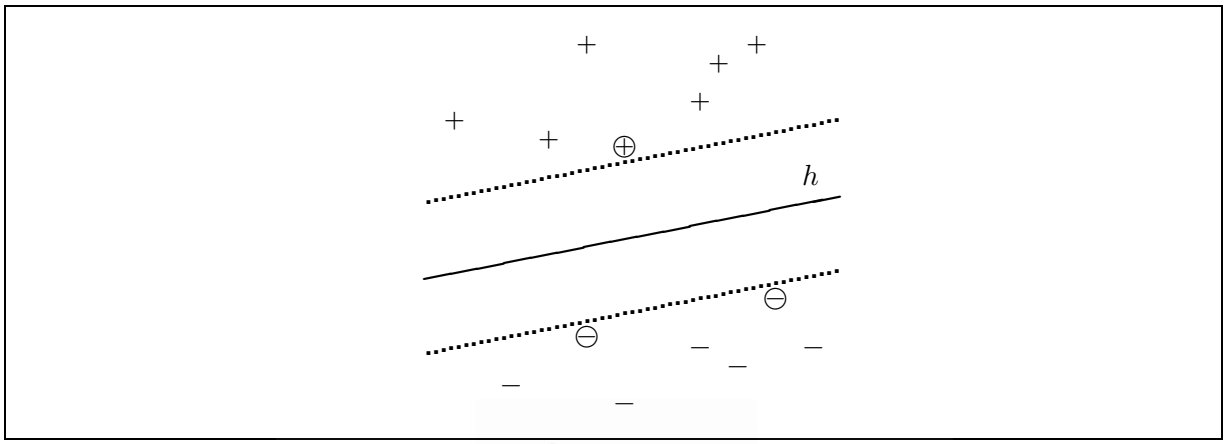


Figura 3.2: Representación gráfica para el concepto de SVM. El entrenamiento de SVM consiste en determinar el hiperplano *h*, que separa los ejemplos positivos y los negativos con un margen máximo. Los ejemplos más cercanos al hiperplano se llaman *Vectores de Soporte* (marcados con círculos).

Universitat d'Alacant
Universidad de Alicante

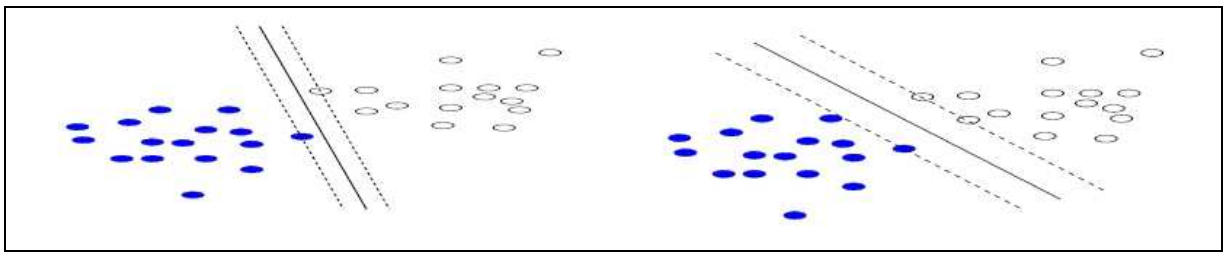


Figura 3.3: Línea de decisión para SVM con margen mínimo (izquierda) y máximo (derecha) [Yang y Liu, 1999].

Una propiedad interesante de SVM es que la superficie de decisión viene determinada sólo por los documentos de entrenamiento que tienen exactamente la distancia $\frac{1}{||\vec{w}||}$ desde el hiperplano. Estos puntos se llaman vectores de soporte (los puntos atravesados por las líneas discontinuas en la figura 3.3) y son los únicos elementos necesarios en el conjunto de entrenamiento para clasificar nuevos documentos. Esta propiedad hace que SVM sea teóricamente muy diferente de otros métodos tales como kNN y Naive Bayes, donde se utilizan todos los documentos de entrenamiento para optimizar la función de decisión.

Es una técnica relativamente nueva y usada por muchos proyectos en los últimos años, como por ejemplo en propuestas de clasificación de páginas web basadas en estructuras jerárquicas [Ceci y Malerba, 2007]. En la actualidad es probablemente el método más adecuado para resolver problemas de clasificación y, por ello, se usa generalmente como clasificador base para comparaciones entre gran cantidad de experimentos que aparecen en la literatura. Sin embargo, hay que tener en cuenta que el planteamiento realizado sólo se puede aplicar a tareas de clasificación binaria, lo que significa que para obtener un ranking de categorías mediante este método, la solución al problema general debe ser abordada como una serie de problemas de clasificación independientes (uno para cada categoría).

A pesar de los prometedores resultados ofrecidos por SVM y de las numerosas optimizaciones que podemos encontrar en la actualidad, sigue presentando la traba de plantear la resolución de un problema de elevada complejidad (la más alta de los algoritmos expuestos en esta sección, tal como se puede consultar en [Yang et al., 2003]). Para la fase de aprendizaje presenta una complejidad cuadrática con respecto al número de documentos de entrenamiento y proporcional al número de términos considerado por categoría. Respecto a la fase de clasificación, el tiempo de respuesta tampoco es despreciable, pues variará linealmente con el número de categorías y el vocabulario de cada una.

Universitat d'Alacant
Universidad de Alicante

3.4.4. Naive bayes (NB)

Los algoritmos de NB se basan en teorías probabilísticas, que permiten estimar la probabilidad de un suceso a partir de la probabilidad de que ocurra otro suceso, del cual depende el primero. Básicamente, se trata de estimar la probabilidad de que un documento pertenezca a una categoría. Dicha pertenencia depende de la posesión de una serie de características, de las cuales conocemos la probabilidad de que aparezcan en los documentos que pertenecen a la categoría en cuestión. Naturalmente, dichas características son los términos que conforman los documentos y tanto su probabilidad de aparición en general, como la probabilidad de que aparezcan en los documentos de una determinada categoría, pueden obtenerse a partir de los documentos de entrenamiento. Para ello se utilizan las frecuencias de aparición en la colección utilizada durante el aprendizaje.

Con dichas probabilidades obtenidas de la colección de entrenamiento, podemos estimar la probabilidad de que un nuevo documento, que contiene un conjunto determinado de términos, pertenezca a cada una de las categorías, pudiendo ordenarlas de forma que el usuario reciba

un ranking de las mismas.

La parte ingenua de este clasificador está en asumir la independencia de las palabras, es decir, que la probabilidad condicional de una palabra dada una categoría se calcula asumiendo que dicha palabra es independiente de las probabilidades condicionales de otras palabras en el documento dada esa categoría. Y aunque la aseveración que se hace es falsa en la mayoría de documentos, este clasificador obtiene sin embargo un rendimiento mayor que otros y nos permite simplificar en gran medida el proceso de clasificación, pues la representación formal de la teoría presentada se reduce a la obtención del producto de las probabilidades de los términos que aparecen en el nuevo documento:

$$P(d_i|c_j) = \prod_k P(w_k|c_j) \ , \tag{3.19}$$

donde $P(w_k|c_j)$ es la probabilidad de que cada palabra w_k que aparece en el documento d_i aparezca también en la clase c_j. Dicha probabilidad se puede calcular fácilmente mediante la obtención de simples estadísticas. Bastará por ejemplo con dividir el número de documentos de entrenamiento de la clase c_j que contienen la palabra w_k entre el número total de documentos dentro del conjunto de entrenamiento.

Acompañando a la evolución de las técnicas para clasificación de documentos se han desarrollado varias versiones de este clasificador [McCallum y Nigam, 1998]. Podemos encontrar principalmente dos modelos (multivariado y multinomial) que usan la misma suposición probabilística. El primero establece representaciones vectoriales binarias indicando si un término aparece o no en un documento, de forma que la presencia o la ausencia de un término contribuyen a la posterior estimación de la probabilidad. El modelo multinomial especifica representaciones de documentos que incluyen el número de apariciones de cada término en los documentos y sólo se tienen en cuenta los términos que aparecen en los documentos de entrenamiento de la clase en cuestión. En la tabla 3.1 se puede encontrar una breve descripción y comparación de ambos métodos.

Los modelos basados en Naive Bayes son muy simples y tienen muy pocos requerimientos computacionales, incluso cuando se utilizan colecciones de datos muy grandes, o vocabularios con gran cantidad de términos, tal como atestiguan numerosos trabajos experimentales [Yang y Liu, 1999; Kim et al., 2000; Diao et al., 2000; Dhillon et al., 2003; Ceci y Malerba, 2007].

Sin embargo, existe un problema muy importante que obliga a ampliar las consideraciones mencionadas mediante versiones más sofisticadas de dicho algoritmo. Cuando se calculan las estadísticas con respecto a una clase, se cuentan las ocurrencias de las palabras en los documentos de entrenamiento. Pero si aparece un nuevo documento que contenga una palabra (o término) que no ha aparecido antes, la probabilidad de dicho término será cero y, consecuentemente, también será cero la probabilidad de pertenencia del documento que la contiene con respecto a cualquier clase, desembocando en resultados erróneos. La cantidad de errores se incrementará cuando se utilicen colecciones de entrenamiento pequeñas, puesto

Modelo Multivariado	Modelo Multinomial
Representa un documento como un vector de palabras, indicando cuáles aparecen y cuáles no.	Representa un documento como un vector de palabras formado sólo por las palabras que aparecen.
No tiene en cuenta el número de apariciones de cada palabra.	Tiene en cuenta la cantidad de ocurrencias de cada palabra.
La probabilidad se calcula multiplicando las probabilidades de cada una de las palabras (aparezcan o no).	La probabilidad se calcula multiplicando las probabilidades de las palabras que aparecen.
Resulta ineficiente para clasificar documentos de tamaños diferentes.	Es más adecuado para conjuntos de documentos con tamaños muy diversos.
Esta técnica ha sido usada desde 1976. Entre los trabajos más representativos se encuentran [Robertson y Sparck Jones, 1988; Lewis, 1992a; Koller y Sahami, 1997].	Esta técnica ha sido usada desde 1994. Entre los trabajos más representativos se encuentran [Lewis y Gale, 1994; Joachims, 1997; Kalt, 1998].

Tabla 3.1: Comparación entre los dos modelos de Naive Bayes. Se puede ampliar dicha información consultando [McCallum y Nigam, 1998].

que si utilizamos un número reducido de muestras, resultará más difícil contemplar a priori cualquier posible palabra.

3.4.5. Clasificadores unificados

Los comités de clasificadores (*classifier committees*) o clasificadores unificados, se basan en el siguiente planteamiento: si para realizar una tarea se requiere un experto, un grupo de expertos llevará a cabo la misma tarea con mejores resultados si la opinión de cada uno se combina de forma adecuada [Sebastiani, 2002].

La idea de los clasificadores unificados es construir un sistema de clasificadores, combinando posteriormente los resultados obtenidos de todos ellos. Esto supone en primer lugar la selección de un número determinado de clasificadores y la decisión de un método para combinar los resultados. Para realizar dicha elección, hay que tener en cuenta que los clasificadores del comité deben ser tan independientes como sea posible, tanto en el procesamiento de documentos como en el método inductivo [Tumer y Ghosh, 1996].

Se supone que cada uno de los clasificadores devuelve un valor CSV (*Classification Score Value*) para cada par documento-categoría. En cuanto a la decisión del método de combinación de estos resultados, son varias las opciones que se han utilizado hasta ahora:

- Por mayoría absoluta: se obtienen valores binarios por cada uno de los clasificadores y se escogerá la opción con más votos (0/1).

- Combinación lineal de los pesos: la suma de los valores CSV de todos los clasificadores da como resultado el valor CSV global, lo que se utiliza para la clasificación final.
- Selección dinámica del clasificador: de todos los clasificadores utilizados, se comprueba cuál de ellos es el más eficiente en la fase de validación y su decisión es la que se toma por válida.
- Combinación adaptativa de clasificadores: se suman las decisiones de todos los clasificadores, pero su valor es ponderado en función del resultado obtenido en la fase de validación por cada clasificador.

Una variante de los clasificadores unificados es el *boosting* [Schapire y Singer, 2000], donde todos los clasificadores aprenden bajo el mismo método de entrenamiento y la unión de éstos se va realizando de forma incremental.

3.4.6. Aproximaciones basadas en recuperación de información

En la sección 2.3.4 ofrecimos detalles sobre la construcción genérica de clasificadores basados en ejemplos, que luego hemos aplicado a algoritmos específicos tales como Rocchio o kNN. Hicimos mención de la necesidad de obtener una medida que nos permita calcular el grado de semejanza de dos documentos y, teniendo en cuenta que hemos representado dichos documentos mediante vectores, podemos formalizar los cálculos fácilmente.

El método más utilizado para evaluar la proximidad de dos vectores es utilizar la medición de la distancia en el espacio euclídeo. Gracias a su simple interpretación geométrica, es la opción preferida por multitud de aplicaciones. Para evaluar la distancia euclídea entre dos documentos, debemos considerar sus vectores como puntos en el espacio m dimensional de la siguiente forma:

$$distancia(\vec{d_1}, \vec{d_2}) = \sqrt{\sum_{j=1}^{m}(w_{1j} - w_{2j})^2} \ , \tag{3.20}$$

donde w_{ij} es el peso del término j en el documento i. El rango de la distancia euclídea va de cero a infinito, tomando el valor nulo cuando los dos vectores son iguales. Sin embargo, al intentar utilizar dicha medida para nuestros propósitos, nos tropezamos con un importante inconveniente: la mayor desventaja de la distancia euclídea aplicada en el ámbito de la clasificación de textos es que dos vectores de documentos pueden presentar un alto grado de similitud aunque no tengan ningún término en común [Willett, 1983]. Y por ello, a pesar de ser un método muy difundido, no se recomienda su uso en clasificadores, donde resulta más común el uso de la métrica del coseno.

Geométricamente, al medir el coseno del ángulo entre dos vectores de documentos, obtenemos una métrica basada en una distancia angular y esto nos permite abstraer variaciones

en la longitud del documento. La similitud se puede calcular mediante el coseno utilizando el producto escalar normalizado:

$$similitud(\vec{d}_1, \vec{d}_2) = \frac{\vec{d}_1 \cdot \vec{d}_2}{||\vec{d}_1|| \cdot ||\vec{d}_2||} = \frac{\sum_{j=1}^{m} w_{1j} \cdot w_{2j}}{\sqrt{\sum_{j=1}^{m} w_{1j}^2}\sqrt{\sum_{j=1}^{m} w_{2j}^2}} \quad . \tag{3.21}$$

En este último caso, el resultado varía en el rango de cero a uno, tomando el valor nulo cuando los vectores están situados perpendicularmente, lo que traducido al mundo real, significa que los dos documentos implicados no tienen ningún término en común.

Conviene mencionar que el cálculo de la similitud mediante el coseno se reduce al simple cálculo del producto escalar si los vectores se encuentran normalizados a longitud unitaria, ya que podemos prescindir del denominador de la ecuación 3.21. Pero aún mediante esta optimización, la complejidad que tendremos que asumir puede resultar relativamente elevada si pretendemos utilizar dicha medida para realizar comparaciones con una gran cantidad de documentos de entrenamiento.

3.5. Corpus y taxonomías

Para poder entrenar un clasificador y evaluar la calidad del proceso de clasificación, necesitaremos un corpus compuesto de una cantidad razonable de documentos ya clasificados y organizados en una serie de categorías. La existencia y libre disposición de un punto de referencia estándar es de gran relevancia, pues permite que cada investigador pueda comparar de forma experimental los resultados de sus propios algoritmos con los resultados obtenidos previamente por otros autores tomando los mismos datos de partida.

De entre todos los corpus que aparecen en la biliografía podemos destacar dos bien diferenciados que cuentan con una amplia difusión: OHSUMED [Hersh et al., 1994] y Reuters [Rose et al., 2002; Debole y Sebastiani, 2004c]. Aunque los describiremos con detalle más adelante, cuando hablemos específicamente del proceso de evaluación, sí vamos a justificar brevemente ahora su uso en otras investigaciones.

3.5.1. OHSUMED

La colección de documentos OHSUMED se creó en la *Oregon Health & Science University* para realizar pruebas en investigaciones de recuperación de información. Está compuesta por un subconjunto de documentos extraídos de la base de datos MEDLINE,[8] cuya elección se orientó al ámbito clínico. Recoge referencias de artículos publicados en gran cantidad de revistas médicas a lo largo de 5 años. Cada documento incluye un título, un resumen, las

[8] Ver http://es.wikipedia.org/wiki/MEDLINE para más información.

categorías que identifican la temática del texto, el autor, la fuente y el tipo de publicación. Puesto que esta colección no está completa ni se encuentra actualizada, no tiene utilidad como herramienta para búsquedas reales y sólo se recomienda su uso para pruebas y evaluación.

3.5.2. Reuters

Reuters es la principal empresa proveedora de información, noticias y tecnología en el sector financiero. Además, es la mayor agencia internacional en publicación de noticias, tanto en texto como en televisión. Cuenta con miles de periodistas, fotógrafos y cámaras, trabajando en decenas de países.

Reuters pretendía crear un conjunto de datos que le permitiera evaluar sus propios sistemas de indexación y clasificación y obtuvo como resultado un recurso único y completamente válido para su uso dentro del ámbito del aprendizaje automático. RCV1 (*Reuters Corpus Volume 1*) es el primer corpus libre y oficial proporcionado por dicha empresa. Contiene casi un millón de noticias en inglés completamente disponibles sin coste alguno. Se encuentran en formato XML estándar y presenta claras ventajas para los investigadores, pues se dispone de diversas herramientas de manipulación y transformación de los documentos. Un aspecto clave de este corpus es el uso exhaustivo de metadatos descriptivos. Todas las noticias se encuentran etiquetadas con códigos de categorías temáticas, regiones geográficas y sectores industriales.

Pero también existen con anterioridad muchas investigaciones que utilizan otras agrupaciones de noticias proporcionadas por Reuters, aunque sobre ellas recaían algunas restricciones de copyright. Por ejemplo podemos hablar de la muy difundida colección de documentos *Reuters-21578*, que se ha empleado para validación de numerosos sistemas de recuperación de información, procesamiento del lenguaje natural y aprendizaje automático. En nuestro caso su uso resulta casi ineludible para poder comparar con trabajos previos, pues además de su amplio uso, se dispone de diversas propuestas de división en conjuntos de entrenamiento y pruebas, que le aportan un valor adicional para la validación de algoritmos de clasificación de textos.

3.6. Escalabilidad

Ya hemos visto que es fácil encontrar una gran variedad de propuestas para seleccionar y ponderar los términos de un corpus para conseguir una representación adecuada de los documentos con el modelo vectorial. También podemos encontrar numerosos planteamientos para conseguir resolver los problemas de clasificación mediante algoritmos que utilicen dichas representaciones. A lo largo de este último capítulo hemos tratado de describir con detalle algunos de dichos estudios. A continuación recapitularemos e intentaremos extraer unas primeras conclusiones sobre las limitaciones de las propuestas existentes que reducen su aplicación final en el mundo real.

Observemos el caso particular del ranking de categorías dentro del ámbito médico, donde cada año se publican miles de nuevos artículos científicos. Para muchos algoritmos de clasificación, el simple hecho de añadir unos cuantos documentos a la colección de entrenamiento repercutiría en la obligatoriedad de tener que repetir completamente el proceso de aprendizaje. Esto es debido a que muchas veces los algoritmos no proporcionan un sistema de entrenamiento incremental. Además, después de cada actualización, la complejidad temporal y espacial final del clasificador suele aumentar considerablemente. Y el problema se agravará si se actualizan las posibles etiquetas a asignar (por ejemplo, por el descubrimiento de una enfermedad que no se encontraba catalogada previamente).

Si el problema a resolver consiste en ordenar un número reducido de categorías cuya temática sea fácilmente distinguible, entonces la solución será fácil de plantear. Pero muchos problemas de clasificación reales dependen de un gran número de categorías, frecuentemente similares. Pensemos por ejemplo en el directorio web *Yahoo! Directory*[9] o el proyecto *Open Directory*,[10] donde nos podemos encontrar con cientos de miles de categorías, agrupadas en jerarquías. Aunque consiguiéramos abordar problemas de tal magnitud con las propuestas planteadas en este capítulo, descubriríamos que no es nada fácil obtener una precisión aceptable, pues ordenar una gran cantidad de categorías que están altamente relacionadas presenta de por sí una dificultad inherente [Manning et al., 2008].

3.6.1. Palabras vs. frases

Intentaremos plantear a continuación algunas de las ventajas e inconvenientes de considerar palabras o frases como características. Por un lado veremos cómo la tradicional bolsa de palabras (BOW) nos puede llevar a soluciones que conllevan una baja complejidad, ofreciendo resultados mucho más que satisfactorios, que hacen que BOW sea la opción más utilizada en la actualidad. Pero por otro lado veremos cómo el uso de frases nos puede aportar ciertos beneficios que muchas veces quedan ocultos por el alto incremento en el tamaño del conjunto de términos a utilizar. Trataremos de justificar de esta forma la necesidad de emplear clasificadores que aprovechen los beneficios que puedan aportar las frases respecto a las palabras, intentando esquivar el problema de la alta dimensionalidad de los vectores de características mediante algoritmos fácilmente escalables.

En una primera aproximación, pueden surgir dos problemas al utilizar la simple bolsa de palabras. Primero, se pierde mucha información lingüística, tal como la secuencia de palabras. Puesto que las secuencias son esenciales para que un ser humano pueda entender el contenido de entidades contextuales, parece razonable pensar que descartarlas podría conducir a una pérdida de información considerable. Un segundo problema es que debemos abstraer distintos factores de relevancia para los diferentes términos que nos encontremos en los documentos. Al utilizar simples palabras, es posible que en algunos casos obtengamos la misma ponderación debido a los pocos matices que contienen. Si incrementamos la sofisticación de las

[9] http://dir.yahoo.com
[10] http://www.dmoz.org

características consideradas, podríamos pensar en cómo cuantificar la importancia de cada una utilizando ciertas particularidades no disponibles anteriormente, de forma que tengamos un indicador más matizado del grado de información representada en cada caso.

Desde un punto de vista intuitivo, una consideración de frases en vez de palabras puede incrementar la calidad y cantidad de información que contienen los vectores de características. Por ejemplo, la expresión "Casa Blanca" pierde su significado si ambas palabras se usan de forma separada. Además, podemos asociar a cada frase sofisticados pesos que contengan alguna información estadística tal como el número de ocurrencias en un documento o dentro del conjunto de entrenamiento. También podemos observar cómo la frase está distribuida entre las diferentes categorías, o incluso podemos extraer ciertos datos intrínsecos de cada expresión, tal como la cantidad o tipo de palabras que contienen.

Las observaciones mencionadas en estos últimos párrafos han conducido a numerosos investigadores a concentrarse en enriquecer el modelo BOW durante muchos años. La mayoría de ellos ha experimentado con n-gramas (n palabras consecutivas) [Mladenić y Grobelnik, 1998; Scott y Matwin, 1999; Tan et al., 2002; Kongovi et al., 2002; Tesar et al., 2006] y otros con conjuntos de items (n palabras que aparecen juntas en un documento) [Meretakis y Wuthrich, 1999; Antonie, 2002; Z. Yang y Zhanhuai, 2003; Tesar et al., 2006]. En algunos casos, se registró un incremento significativo del rendimiento, pero muchas veces sólo se apreciaba una ligera mejoría o incluso un leve empeoramiento en los resultados.

Pero sea cual sea el rendimiento que obtengamos, una conclusión clara a la que llegaremos si optamos por emplear frases en vez de palabras para representar los documentos, es que nos tendremos que enfrentar a un inevitable incremento en la dimensionalidad de los vectores de características. Si no potenciamos otros aspectos del clasificador, mejorando por ejemplo la selección y ponderación de los términos, probablemente será mejor emplear el sistema BOW, que actualmente es la opción más utilizada.

Universitat d'Alacant
Universidad de Alicante

3.6.2. Dimensionalidad y cálculo de pesos

Las medidas IG y χ^2 se pueden utilizar para reducir la dimensionalidad del espacio vectorial en un factor de 100 sin apreciar pérdida de la eficiencia en la clasificación [Yang y Pedersen, 1997; Forman, 2003]. Una vez limitado el número de características, podemos proceder a ponderarlas por ejemplo utilizando OR, ConfWeight o RF. Pero nos topamos con el problema de que todas estas medidas presentan una complejidad temporal cuadrática, que es proporcional al número de categorías (C) y al número total de términos en el conjunto del vocabulario (V): $O(CV)$ [Yang y Pedersen, 1997]. Esto es debido a que los cálculos se repiten para cada una de las categorías del problema a resolver, pues dependen de la pertenencia o no pertenencia de los documentos a cada una de las clases. De hecho, la intuición en la que se basan es que los términos mejor valorados son aquellos que se encuentran más distribuidos entre las muestras positivas y las negativas.

Por otro lado, las frecuencias de término y documental asumen simplemente que una vez

eliminadas aquellas palabras que no aporten ningún significado (por ejemplo, mediante el uso de stopwords), los términos más representativos del contenido de un texto serán aquellos que más aparezcan en él y aquellos muy comunes (por su aparición en muchos documentos) o bien no proporcionarán información para predecir una categoría, o simplemente no tendrán repercusión en el rendimiento global. De esta forma, se puede establecer que el valor de un término es proporcional a la frecuencia del término en el documento (TF) e inversamente proporcional a su frecuencia en el corpus (IDF). Este sencillo planteamiento anticipa la ventaja que plantean estas últimas medidas, pues conlleva que se puedan obtener fácilmente, ya que sólo es necesario contabilizar las apariciones totales de cada término y las ocurrencias particulares para cada documento [Apté et al., 1994a; Yang y Pedersen, 1997; Forman, 2003]. Además, tal como se observa en la ecuación 2.1, estos valores se pueden combinar rápidamente en uno solo ($TF \cdot IDF$) que nos permite ponderar cada término con una sola medida.

3.6.3. Vectores de soporte, vecinos y prototipos

Para cada posible categoría, durante el proceso de aprendizaje, SVM intenta generar dos zonas en el conjunto de vectores para separar las muestras positivas de las negativas, lo que nos lleva a la resolución de un problema de relativa dificultad. Aunque finalizado este proceso, durante la fase de clasificación bastará con decidir en qué zona se sitúa el nuevo documento para deducir la clase de pertenencia. Pero aún ofreciendo unos resultados comparables a los obtenidos por humanos y, a pesar de disponer de numerosas optimizaciones,[11] su aplicación se encuentra obstaculizada porque parte inicialmente con una complejidad muy elevada: $O(CD^{1,7}M_{ave})$, donde C es el número de categorías, D el número de documentos de entrenamiento y M_{ave} el número medio de características por documento. Con respecto a la fase de clasificación, la parte más difícil consiste en calcular el producto escalar entre el nuevo vector de entrada (documento) y los vectores de soporte de cada clase, lo que nos conduce a una complejidad $O(CM_a)$, donde M_a se refiere a la longitud del documento de prueba, es decir, al número de características que contiene.

A diferencia de SVM, una de las ventajas de los clasificadores basados en la regla de los k-vecinos más cercanos es la carencia de un proceso específico de aprendizaje. Pero como contrapartida, cada vez que el algoritmo recibe un nuevo documento, éste se debe comparar con cada una de las muestras de entrenamiento para poder emitir una decisión. De esta forma, la complejidad temporal para completar el entrenamiento es casi despreciable.[12] Para clasificar un nuevo documento la complejidad que obtenemos es lineal con respecto al tamaño de la colección de entrenamiento (D) y con el número medio de términos que contienen los documentos de entrenamiento (L_{ave}) y de prueba (M_a), resultando en $O(DL_{ave}M_a)$.

[11] Las últimas investigaciones demuestran empíricamente que la complejidad requerida para entrenar SVM puede llegar a ser del orden de $O(D^{1,7})$ (consultada en [Joachims, 2006]), aunque el problema matemático en el que se basa su resolución presenta complejidad cúbica [Kozlov et al., 1979].

[12] En kNN el único entrenamiento necesario consiste en un preproceso para extraer los tokens de cada documento (una fase que requieren el resto de clasificadores) y establecer un valor óptimo de k, aunque éste se puede fijar manualmente, llevando a un tiempo de aprendizaje nulo.

Como se puede observar, para un número de muestras (D) muy grande, kNN puede resultar verdaderamente ineficiente.

Por otro lado, nos encontramos con los algoritmos tipo Rocchio, cuyo proceso de aprendizaje, aunque obligatorio, implica simplemente el promediado de los vectores de los documentos de entrenamiento. Estos algoritmos nos ofrecen la ventaja de reducir en gran medida el tiempo de respuesta, pues para clasificar un nuevo documento, en vez de compararlo con cada documento ya clasificado, sólo deberemos calcular su similitud con el prototipo representativo de cada categoría. Para completar el aprendizaje deberemos realizar el cálculo de la media, sumando y dividiendo los vectores de entrenamiento. La complejidad temporal para calcular la suma de todos los pesos de los vectores de cada categoría es lineal con el número de ejemplos de entrenamiento (D) y la cantidad media de características por documento (L_{ave}), puesto que sólo tenemos que considerar aquellos términos que tengan un valor distinto de cero. El tiempo requerido para calcular la división de cada uno de estos vectores suma varía linealmente con el número de categorías (C) y el número de términos en el vocabulario (V). Si juntamos estos dos valores, nos conducen a $O(DL_{ave} + CV)$. Para clasificar un nuevo documento necesitamos calcular la distancia euclídea entre los prototipos de cada clase y el documento de prueba, por lo que el tiempo requerido es proporcional al número de categorías (C) y la cantidad de términos en el documento de prueba (M_a). Así, el tiempo de ejecución viene indicado por $O(CM_a)$, valor que no se verá afectado a medida que incrementemos el número de muestras de entrenamiento del corpus y que, en consecuencia, en la mayoría de los casos resultará bastante menor que el valor ofrecido por kNN y siempre menor que SVM.

En la tabla 3.2 podemos encontrar comparaciones de la complejidad que presentan diversos métodos de clasificación. Aunque algunos clasificadores Bayesianos presentan una complejidad lineal, otros muchos algoritmos tienen una complejidad cuadrática y tampoco ofrecen la posibilidad de realizar un aprendizaje incremental, lo que dificulta su uso en el mundo real. Con respecto a SVM, la mayoría del trabajo reciente se ha dedicado a reducir su complejidad, muchas veces mediante soluciones basadas en aproximaciones al problema original. En cualquier caso, el tiempo requerido por las soluciones tradicionales para completar el aprendizaje dificulta (o imposibilita) su uso con colecciones de entrenamiento muy grandes. Otras alternativas que ofrecen complejidad lineal respecto al número de documentos empleado, no presentan una buena escalabilidad al incrementar el número de características [Manning et al., 2008]. Por ejemplo, dentro del ámbito médico, según [Yang et al., 2003] la implementación "escalable" del algoritmo SVM propuesta en [Joachims, 1999] necesitaría unas 102 horas para entrenar un clasificador con el propósito de discriminar entre las aproximadamente 15.000 categorías incluidas en el MESH.[13] Considerando estas cifras y utilizando la tecnología computacional de esa época, si hubiéramos intentando utilizar dicho algoritmo para resolver problemas de clasificación dentro del campo de la ontología genética,[14] con varios cientos de miles de conceptos, se habrían requerido unos 4 meses para obtener los primeros resultados. Además, hay que tener en cuenta que para realizar estos cálculos hemos contabilizado sólo el incremento

[13] Medical Subject Headings (http://www.nlm.nih.gov/mesh).
[14] http://www.geneontology.org

Clasificador	Modo	Complejidad temporal
Naive Bayes	entrenamiento	$O(DL_{ave} + CV)$
Naive Bayes	prueba	$O(CM_a)$
Rocchio	entrenamiento	$O(DL_{ave} + CV)$
Rocchio	prueba	$O(CM_a)$
kNN	entrenamiento	$O(1)$
kNN	prueba	$O(DL_{ave}M_a)$
SVM	entrenamiento	$O(CD^{1,7}M_{ave})$
SVM	prueba	$O(CM_a)$

Tabla 3.2: Complejidad temporal de diversos clasificadores [Manning et al., 2008]. Con *entrenamiento* nos referimos al tiempo necesario para completar el aprendizaje del clasificador utilizando D documentos. La fase de *prueba* se realiza sobre un solo documento. C es el número de categorías y V es el número de elementos en el conjunto del vocabulario. L_{ave} es el número medio de tokens por documento, mientras que M_{ave} es el número medio de características por documento que forman parte del vocabulario (por ejemplo, serían los tokens que quedan después de haber eliminado las palabras de parada). L_a y M_a son los equivalentes en el documento de prueba.

Corpus	Naive Bayes	Rocchio	kNN	linear SVM	rbf SVM
WebKB	82,0	74,1	80,5	91,6	90,1
Reuters-21578	72,3	79,9	82,6	87,5	86,4
OHSUMED	62,4	61,5	63,4	71,6	69,8

Tabla 3.3: Precisión medida con F1 entre distintos corpus y clasificadores (ordenados por complejidad de forma ascendente de izquierda a derecha) [Joachims, 2002].

en el número de categorías, no en la cantidad de documentos de entrenamiento.[15]

A pesar de estas últimas cifras, SVM sigue siendo uno de los algoritmos más referenciados en la bibliografía. En muchas ocasiones se suele utilizar como clasificador base, para evaluar si los resultados de nuevos métodos propuestos proporcionan la suficiente validez. En la tabla 3.3 podemos comparar el rendimiento de los clasificadores mencionados, donde se puede observar que en la mayoría de los casos el incremento de la complejidad tiene la recompensa de una mejoría en la calidad de los resultados. En las últimas columnas se sitúa SVM, cuya efectividad sigue siendo hoy en día la meta a alcanzar. Sin embargo, una alta precisión no resulta útil si no viene acompañada de una alta aplicación en el mundo real. En los próximos capítulos trataremos de proporcionar un método altamente escalable e incremental que intente abordar ambos objetivos, de forma que finalmente pueda ser utilizado para la categorización de corpus con un elevado número de categorías y documentos de entrenamiento.

[15] En [Ruch, 2005] se estima un tiempo de aprendizaje de 20 años.

Universitat d'Alacant
Universidad de Alicante

Capítulo 4

Ponderación y selección de características para la clasificación de textos

A lo largo de esta tesis exponemos algunas consideraciones a tener en cuenta para conseguir resolver de forma eficiente problemas relacionados con el ranking de categorías y la clasificación de textos. Más concretamente, pretendemos demostrar que una combinación adecuada de ciertos parámetros puede mejorar el tiempo de respuesta y la precisión presentados en otros trabajos publicados previamente.

En las próximas secciones planteamos un método automático de selección y ponderación de características utilizando algunos conceptos nuevos y otros que han sido estudiados en el pasado, tales como el uso de sintagmas nominales, meta-información, etiquetado de partes de la oración, palabras de parada y reducción de dimensiones, entre otros. Como vimos en el capítulo anterior, muchos de estos conceptos se usan habitualmente [Scott y Matwin, 1999; Basili et al., 2000; Granitzer, 2003; Moschitti y Basili, 2004], sin embargo, suelen emplearse para alimentar complejos algoritmos de clasificación que no suelen aprovechar adecuadamente tales esfuerzos por optimizar los datos de entrada. Nosotros intentamos potenciar y matizar ciertos aspectos de forma que, eligiendo y combinando adecuadamente ciertas particularidades de las expresiones contenidas en los textos, sea posible crear un sistema de valoración de términos para obtener un método eficiente de clasificación que permita reemplazar a otros algoritmos más complejos, como SVM, ya descrito en anteriores capítulos.

A continuación explicaremos las intuiciones que nos han conducido a desarrollar los experimentos en cuestión y expondremos los detalles de la investigación, proporcionando una descripción completa del algoritmo. En primer lugar analizaremos las peculiaridades del nuevo método propuesto de selección y ponderación de características. Una vez procesado todo el conjunto de documentos de entrenamiento y habiendo obtenido todos los términos a considerar, intentaremos reducir dicho conjunto seleccionando sólo las características más relevantes. Por último, veremos cómo obtener prototipos representativos de categorías y cómo calcular

la similitud entre dichos prototipos y los vectores resultantes de procesar nuevos documentos.

4.1. Características de aprendizaje

En esta sección proponemos la inclusión de ciertos parámetros y consideraciones en el proceso de transformación de los documentos en vectores de características. Intentaremos incrementar la calidad de representación para conseguir mejorar el resultado de la clasificación y en particular del ranking de categorías. Agruparemos algunas de las aproximaciones ya presentadas en estudios previos y también realizaremos nuevas aportaciones, justificando el porqué de nuestros planteamientos.

Para tratar de detectar, aislar y extraer los conceptos que contienen los textos del corpus, vamos a utilizar los sintagmas nominales (*noun phrases* [Scott y Matwin, 1999]) como un tipo particular de n-gramas que están compuestos exclusivamente de sustantivos que pueden aparecer o no precedidos por otros sustantivos o adjetivos. Dichos n-gramas formarán las características individuales que constituyan las dimensiones de los vectores de documentos, lo que incrementará la capacidad representacional de nuestra solución con respecto a la aproximación basada en bolsa de palabras. Trataremos de argumentar las ventajas que obtenemos, pues por ejemplo podremos disponer de propiedades adicionales, como la longitud de los términos, que podrá usarse como indicador de la cantidad de información representada.

4.1.1. Descriptores de categorías

Las colecciones de documentos utilizadas para el aprendizaje de los clasificadores supervisados se encuentran organizadas en una serie de categorías preestablecidas de forma manual. Esto es, para poder entrenar el algoritmo de clasificación, un ser humano ha debido asignar previamente según su criterio una serie de documentos a una o más clases temáticas dependiendo del corpus con el que se esté trabajando. Para ayudar en este proceso y para asegurar que diferentes expertos utilizan los mismos criterios, es habitual identificar unívocamente todas las categorías mediante una serie de términos clave (*descriptores de categorías*) que establecen los temas de los que tratan los documentos que pertenecen a cada clase en particular.

Por lo tanto, parece lógico pensar que deberíamos buscar en los documentos los términos utilizados en los descriptores, pues ya han sido valorados por expertos y su presencia indica un alto grado de pertenencia a la categoría que identifiquen.

4.1.2. Sustantivos y adjetivos

Existen ciertas categorías gramaticales que no tienen una aportación importante para tareas de clasificación (por ejemplo, artículos o preposiciones) porque el contenido semántico de un texto viene representado generalmente por los sintagmas nominales [Scott y Matwin,

1999]. Además, si echamos un vistazo a los descriptores de las categorías, se puede observar que la mayoría de ellos están compuestos únicamente por sustantivos y adjetivos.

Debemos asumir que es casi imposible detectar todos los sintagmas nominales. Además, los corpus técnicos se actualizan continuamente con nuevas palabras y abreviaturas, las cuales muchas veces no se pueden detectar por el contexto, puesto que no se incluyen en sentencias que se den con frecuencia en el idioma. Nosotros sugerimos que esos términos desconocidos sean considerados sustantivos, puesto que son implícitamente poco comunes y discriminativos.

Algunas de las técnicas asociadas al tratamiento de sustantivos y adjetivos como características de aprendizaje son la detección de partes de la oración, la obtención de raíces, el uso de palabras y términos, así como la eliminación de palabras de parada.

Partes de la oración y raíces

Podemos encontrar multitud de herramientas para identificar las partes de la oración (*part-of-speech tagging* o *PoS tagging*) y obtener las raíces de los términos (*stemming*). Con la mayoría se puede obtener una alta precisión, pero no dejan de ser algoritmos que implican cierto coste computacional y, como consecuencia, incrementan el tiempo necesario para completar el procesamiento de los documentos. Utilizando un simple diccionario podemos identificar fácilmente los posibles sustantivos y adjetivos, y los podemos reducir al masculino singular en un solo paso, empleando un mínimo tiempo de respuesta. Incluso se puede optar por una solución mixta, que nos permitirá optimizar la precisión de los resultados, pues se pueden utilizar un etiquetador y un diccionario al mismo tiempo, considerando por ejemplo como palabras válidas aquellas identificadas mediante cualquiera de los dos métodos.

Universitat d'Alacant
Universidad de Alicante

Palabras y términos

Si agrupamos una palabra junto con sus palabras adyacentes (una frase) para construir las características que forman el perfil de cada categoría, podemos llegar a incrementar el poder discriminador de las mismas. Esto se debe a que el empaquetamiento de palabras puede aportar valor semántico y también permite filtrar palabras que aparecen frecuentemente al utilizar expresiones más complejas que representan de forma particular a la categoría [Kongovi et al., 2002].

Además, puede resultar muy útil agrupar las palabras de forma que el número de caracteres en una secuencia de palabras, o lo que es lo mismo, la longitud de cada término (*Term Length - TL*), pueda ser usada como una medida de relevancia indicativa de la cantidad de información que contiene. De esta forma, podemos considerar que las expresiones más largas serán más relevantes.

Términos no descriptivos

Tal como se vio en la sección 3.2.1, la presencia de términos neutros carentes de información semántica se puede evitar usando una lista fija de palabras de parada (*stopwords*). Sin embargo, si sólo disponemos de un listado genérico, algunos términos válidos podrían ser eliminados erróneamente al trabajar con corpus que abarquen gran cantidad de temas. Por el contrario, si utilizamos una lista muy reducida, podríamos dejar de filtrar términos muy comunes para un determinado tipo de documentos. En las próximas secciones mostraremos cómo construir esta lista automáticamente y, además, probaremos que el utilizar una lista adaptada al corpus puede resultar conveniente, ya que no sólo conseguimos mejorar el rendimiento, sino la precisión de los resultados obtenidos durante el proceso de clasificación.

4.1.3. Selección de términos válidos

Como es habitual en el proceso de entrenamiento, en un primer paso debemos extraer el conjunto de todas las características potenciales. En nuestro caso, serán todas aquellas expresiones que contengan sustantivos y adjetivos, sin preocuparnos en principio de la calidad o cantidad de términos obtenidos. Posteriormente, seleccionaremos de entre todos ellos los más relevantes empleando una medida de discriminación media como umbral de corte.

Para llevar a cabo un aprendizaje adecuado es necesario procesar corpus de tamaño considerable, con el propósito de obtener una representación que recopile suficiente información de los datos de entrada de forma que pueda ser utilizada para alimentar al clasificador. Esto se consigue mediante diversas transformaciones que se aplican repetitivamente sobre cada documento y que proporcionan finalmente una serie de vectores de características que representarán el contenido de los mismos. Con el objetivo de agilizar dicho proceso, debemos buscar algún método simple y eficaz que nos permita analizar los textos para conseguir detectar y extraer términos válidos. Siguiendo una serie de sencillos pasos podemos conseguirlo de la siguiente forma:

1. Sólo los sustantivos y adjetivos se toman en consideración. Cualquier otra parte de la oración (verbos, preposiciones, conjunciones, etc.) se descarta. Para este propósito, se pueden usar un etiquetador[1] (PoS tagger) y un diccionario.[2] Las palabras que no se encuentran en el diccionario, se consideran sustantivos.

2. Las palabras se reducen a sus raíces. Puesto que cada característica estará formada por términos compuestos sólo de sustantivos y adjetivos, no necesitamos aplicar un proceso específico de stemming, ya que basta con eliminar el género y el número de cada palabra. Teniendo en cuenta además que todos los documentos del conjunto de aprendizaje están escritos en inglés, sólo necesitamos consultar de nuevo el diccionario, esta vez

[1] Hemos empleado la herramienta SVMTool [Màrquez y Giménez, 2004] como etiquetador.

[2] El diccionario que utilizamos en esta tesis se puede descargar de www-formal.standford.edu/jsierra/cs193l-project/morphological-db.lisp.

para reducir cada palabra a su forma masculino singular. Por ejemplo, las expresiones "Children's Diseases" y "child's disease" apuntarían a la misma característica.

3. De cada oración se extraen aquellos términos que estén compuestos de secuencias de sustantivos o adjetivos que terminen con un sustantivo. Utilizando una expresión regular, esto se podría representar como "*[Adjetivo, Sustantivo]* Sustantivo*". Esta consideración viene motivada por el hecho de que las expresiones más útiles son aquellas que identifican los objetos y los sujetos de una oración [Scott y Matwin, 1999]. Además, utilizando este tipo específico de n-gramas, podemos incrementar la capacidad de representación respecto a la simple bolsa de palabras, sin elevar excesivamente el número de características generadas. Por ejemplo, los términos extraídos de "*Ultrasound examinations detect cardiac abnormalities*" son:

ultrasound	*cardiac abnormality*
ultrasound examination	*abnormality*
examination	

Una vez concluido, este proceso nos proporcionará el conjunto de términos válidos (*Valid Terms - VT*) de las colecciones de documentos de entrenamiento de cada categoría. Tales expresiones se encuentran formadas por secuencias de palabras que constituyen los elementos indivisibles que serán utilizados como características. Llegados a este punto ya podemos observar que las palabras "término" y "característica", en nuestro caso se refieren al mismo concepto.

4.2. Ponderación de características

Las características descritas anteriormente son sólo el resultado de un primer paso. Aunque ya disponemos de una serie de conjuntos de términos, cada uno representando e identificando una categoría específica, todavía necesitamos una valoración numérica que nos permita determinar la importancia de cada término (peso) en cada una de las clases del dominio de trabajo. Además, debido a la gran cantidad de expresiones obtenidas, resulta altamente recomendable aplicar alguna técnica de reducción de dimensionalidad con el propósito de seleccionar sólo aquellos términos que mejor identifiquen cada categoría. A continuación procederemos a enumerar varios datos estadísticos que más adelante utilizaremos siguiendo ciertos criterios y que nos permitirán llevar a cabo una valoración y selección de términos.

Para ponderar los términos obtenidos al procesar los documentos, muchos métodos proponen utilizar la frecuencia del término (*Term Frequency - TF*) y la frecuencia del documento (*Document Frequency - DF*) calculadas sobre el corpus completo. De esta forma, los términos más importantes son aquellos que aparecen frecuentemente y que tienen una naturaleza discriminativa por su aparición en sólo unos pocos documentos. Nosotros vamos a combinar el uso de estas medidas tomadas sobre el corpus completo, junto con su uso como indicadores de pertenencia a nivel de clase, puesto que aquellos términos con valores altos para una

categoría en particular, serán mejores candidatos para identificar la materia que representan los documentos que los contienen. Además, tal como veremos en los siguientes apartados, proponemos el uso de factores adicionales a la frecuencia de los términos, como pueden ser la localización de los mismos dentro del documento (título o resumen).

4.2.1. Localización dentro del documento

En la sección 2.4 argumentábamos que una de las aplicaciones más difundidas de la clasificación de textos es la organización de la literatura científica en bases de datos bibliográficas. Para este propósito, los autores deben identificar sus publicaciones mediante un título que resuma brevemente el contenido del documento en una sola oración, siguiendo un procedimiento similar al de otras situaciones, como en el caso de los correos electrónicos, donde se dispone de un campo de "asunto" para tal fin. Esta información resulta de gran relevancia, pues nos indica qué términos pueden llegar a condensar los conceptos que se pretenden explicar. Sin embargo, muchos modelos de clasificación ni siquiera lo tienen en cuenta, como es el caso del esquema TFIDF, donde las expresiones se valoran independientemente del lugar donde se encuentren. Incluso en algunas ocasiones se descartaría un término que aparezca sólo una vez en un título porque su relevancia no se pueda confirmar. Por ejemplo, tal como se expuso en la sección 3.2.1, suele ser habitual utilizar un umbral de 3 sobre la frecuencia del documento [Granitzer, 2003], de forma que los términos que no aparecen en al menos tres documentos se descartan antes del proceso de aprendizaje. Por ello, en muchas situaciones la ponderación de términos o la reducción de la dimensionalidad por dichos criterios conlleva despreciar cierta información disponible en la mayoría de los documentos a clasificar. Nosotros planteamos considerar de forma diferente los términos que aparezcan en los títulos, dándoles más importancia, pues ésa era la intención del autor.

Universidad de Alicante

4.2.2. Particularización por categoría

La sencillez del esquema TFIDF y la eficiencia en los resultados presentados en la literatura [Joachims, 1998; Sebastiani, 2002; Granitzer, 2003], lo han convertido en uno de los métodos de ponderación más populares. Sin embargo, utilizando esta medida no se tiene en cuenta la clase a la que pertenecen los documentos en los que aparece el término, desperdiciando una información potencialmente relevante por una traba de coste computacional. La forma de incluir esta información en los vectores de características puede simplificarse, evitando el aumento en la complejidad del proceso de aprendizaje respecto a la ponderación básica mediante TFIDF. Veremos que dicha información también se puede abstraer fácilmente mediante la obtención de simples estadísticas, tal como se demostrará más adelante, en la sección 4.2.4.

También hay que tener en cuenta que si obtenemos un conjunto específico del vocabulario para cada categoría, tendremos a nuestro alcance una nueva medida que evalúa la distribución de los términos según si han sido considerados relevantes para identificar una clase en particular. Este nuevo concepto, que denominaremos Frecuencia de Categorías (*Category*

Frequency - CF) nos servirá para calcular la capacidad discriminativa de los términos con mayor efectividad que otras medidas tradicionales, ya que además de evaluar la aparición de las características en todo el corpus o en las muestras positivas y negativas, podemos tener en cuenta la representatividad local con respecto a cada categoría. Más concretamente, CF_j mide el número de categorías en las que un determinado término t_j aparece. Por lo tanto, valores grandes de CF_j indicarán menor capacidad discriminativa de t_j.

4.2.3. Normalización de pesos

Tal como vimos en la sección 2.3.3, los documentos grandes contienen numerosos términos diferentes, que aparecen habitualmente de manera repetida. De esta forma, se incrementa la contribución media de sus características en favor de la similitud hacia nuevos textos y, por lo tanto, adquieren preferencia sobre documentos más pequeños [Singhal et al., 1996].

Se ha demostrado que la normalización de los pesos de cada característica es una técnica efectiva para solucionar el problema de diferencias de tamaño entre los distintos documentos [Buckley, 1993]. Cuando intentamos obtener un ranking de categorías, también nos debemos enfrentar al hecho de que las diferentes clases no se encuentran igualmente pobladas, y por lo tanto, también es probable que las primeras posiciones en los resultados caigan sobre las categorías que tienen mayor número de muestras. Por ello puede resultar conveniente aplicar algún factor de normalización sobre las medidas TF y DF de forma que no dependan de la cantidad de documentos de entrenamiento de cada clase, pudiendo ser usadas consecuentemente no sólo como métricas de ponderación sino como indicadores de pertenencia que nos permitan realizar comparaciones entre clases directamente.

Universitat d'Alacant
Universidad de Alicante

4.2.4. Cálculo de TF, DF y CF

Puesto que partimos de un proceso de aprendizaje supervisado, disponemos de la información de pertenencia de cada documento a varias categorías. Si abordamos el proceso de aprendizaje como si estuviéramos diseñando un clasificador específico para cada clase y utilizamos en cada caso su propio conjunto de entrenamiento, obtendremos un modelo que relaciona intrínsecamente la presencia de ciertos términos con el área temática de la que trate el texto donde se hayan encontrado. Por ejemplo, podemos hallar palabras muy frecuentes dentro de una categoría en particular que, sin embargo, no aparezcan en ninguna otra clase. Obviamente, la presencia de tales términos se verá fuertemente asociada al tema del que traten los documentos donde se encuentran y, por lo tanto, serán altamente discriminativos y relevantes.

El objetivo no es realizar un entrenamiento como el empleado por los clasificadores de una clase (*one-class classifiers*) [Tax y Duin, 2001], clasificadores muy específicos que indican si una muestra pertenece a una clase o no. Nuestra propuesta se basa en calcular ponderaciones diferentes para cada término, con el propósito de que cada característica tenga un peso

diferente para cada categoría.

Considerando el planteamiento expuesto, nuestro punto de partida son m colecciones de documentos, formadas cada una con los documentos pertenecientes a cada categoría. Los subconjuntos se procesan por separado para calcular las frecuencias de aparición de los términos, que se pueden obtener mediante un sencillo recorrido estadístico. De esta forma, todos los valores requeridos por el algoritmo para obtener los pesos finales se podrían colocar en una matriz donde las columnas corresponden con términos y las filas con categorías:

	t_1	t_2	$\cdots$	t_n	
c_1	TF_{11}, DF_{11}	TF_{12}, DF_{12}	$\cdots$	TF_{1n}, DF_{1n}	N_1
c_2	TF_{21}, DF_{21}	TF_{22}, DF_{22}	$\cdots$	TF_{2n}, DF_{2n}	N_2
$\cdots$	$\cdots$	$\cdots$	$\cdots$	$\cdots$	$\cdots$
c_i	TF_{i1}, DF_{i1}	TF_{i2}, DF_{i2}	$\cdots$	TF_{in}, DF_{in}	N_i
$\cdots$	$\cdots$	$\cdots$	$\cdots$	$\cdots$	$\cdots$
c_m	TF_{m1}, DF_{m1}	TF_{m2}, DF_{m2}	$\cdots$	TF_{mn}, DF_{mn}	N_m
	CF_1	CF_2	$\cdots$	CF_n	

donde:

- $c_i =$ categoría *i*.
- $t_j =$ término *j*.
- $n =$ número de términos extraídos de todos los documentos de aprendizaje del corpus completo.
- $m =$ número de categorías.
- $TF_{ij} =$ Frecuencia del término t_j, es decir, número de veces que el término t_j aparece en todos los documentos de aprendizaje para la categoría c_i.
- $DF_{ij} =$ Frecuencia de documento del término t_j, es decir, número de documentos de aprendizaje para la categoría c_i en los que aparece el término t_j.
- $CF_j =$ Frecuencia de categoría del término t_j, es decir, número de categorías en las que aparece el término t_j.
- $N_i =$ Número de términos extraídos de los documentos de aprendizaje de la categoría c_i.

Finalmente, cada CF_j y N_i puede ser calculado fácilmente a partir de TF_{ij}, pues sólo hay que tener en cuenta si el término aparece o no, es decir, si su frecuencia es cero o diferente de cero. Más formalmente:

$$CF_j = \sum_{i=1}^{m} x_{ij} \ ; \ N_i = \sum_{j=1}^{n} x_{ij} \ ; \ x_{ij} = \begin{cases} 1 & \text{si } TF_{ij} \neq 0 \\ 0 & \text{en otro caso} \end{cases} . \tag{4.1}$$

Universitat d'Alacant
Universidad de Alicante

disease (50, 9)	*carcinoid heart (26, 8)*	*case (10, 2)*
heart (48, 9)	*carcinoid heart disease (26, 8)*	*level (9, 3)*
carcinoid (44, 8)	*valve (16, 6)*	*carcinoid syndrome (8, 3)*
heart disease (40, 9)	*plasma (14, 3)*	*syndrome (8, 3)*
patient (32, 6)	*tricuspid valve (11, 5)*	...

Figura 4.1: Términos válidos del corpus OHSUMED, sub-árbol de enfermedades del corazón, categoría "Carcinoid Heart Disease".

Todavía podemos realizar una consideración adicional que nos permita llevar a la práctica la idea expuesta en la sección 4.2.1, donde hacíamos referencia a cómo un autor destaca en el título de un artículo las ideas principales que se van a exponer. Si tratamos de forma diferente las frecuencias obtenidas de los términos que aparecen en los títulos, probablemente consigamos mejorar la capacidad informativa de las estadísticas generadas. Después de realizar algunos experimentos para averiguar cuál es el factor más adecuado que nos permita incrementar su valoración,[3] hemos detectado que duplicar los valores de TF y DF al procesar los títulos de los documentos consigue optimizar la precisión en los resultados.

Para que podamos observar algunas de las expresiones y frecuencias generadas a partir de textos reales, a lo largo de este trabajo mostraremos algunos ejemplos con las expresiones obtenidas del proceso de aprendizaje. Indicaremos las medidas TF y DF correspondientes a cada término entre paréntesis en la forma (TF, DF). Esto nos ayudará a plasmar mejor el proceso de selección de términos válidos, viendo cómo contabilizamos al mismo tiempo sus frecuencias de aparición. Por ejemplo, en la tabla 4.1 podemos observar el resultado que obtenemos después de analizar los documentos asociados con la categoría "Carcinoid Heart Disease" (corpus OHSUMED[4]), mostrando algunas de las expresiones válidas seleccionadas con mayor número de repeticiones. Observamos, por ejemplo, el término *tricuspid valve (11,5)*, que muestra como *tricuspid valve* aparece 11 veces en 5 de los documentos de entrenamiento de la categoría "Carcinoid Heart Disease".

Universidad de Alicante

4.2.5. Cálculo de la relevancia

En este punto, ya disponemos de los términos más relevantes para cada una de las m categorías del conjunto de aprendizaje. Sin embargo, no todos los términos tienen el mismo valor representativo y discriminatorio. Por ello, el siguiente paso debe consistir en ponderar todos los términos para medir la importancia relativa de cada una dentro de las categorías.

En los próximos párrafos vamos a formalizar las intuiciones planteadas en la sección 4.1 mediante una agrupación de los parámetros que allí expusimos y que nos permitirán establecer una medida de relevancia (peso). En primer lugar recordemos que hemos extraído un conjunto individual de características por cada categoría con el propósito añadido de obtener también pesos diferentes para los mismos términos en categorías diferentes. El objetivo es calcular

[3] Consultar el capítulo de evaluación, sección 5.4.3 página 133 para más detalles.
[4] Se puede consultar la sección 5.1, página 105 para más detalles sobre el corpus OHSUMED.

ciertos pesos w_{ij} cuyo valor sea mayor para aquellos términos t_j que sean más representativas para cada clase c_i. Tendremos en cuenta la longitud de los términos, y las frecuencias de aparición en cada documento y en toda la colección de textos de entrenamiento, así como su distribución entre las diferentes clases. Partiendo de dichas medidas proponemos una forma de ponderación sencilla, como la utilizada por el esquema TFIDF, pero más informativa y que cumple al mismo tiempo con el perfil esperado para el diseño genérico de cualquier clasificador.

Tal como se expuso en la sección 2.3.3, ecuación 2.1, el peso de cada término depende habitualmente de tres factores: local, global y normalización. A continuación formulamos el procedimiento para obtener cada uno de estos valores.

Factor local y factor global

Respecto al factor local, seguiremos el mismo criterio que muchos de nuestros predecesores y procederemos a contabilizar los términos para valorar la cantidad de repeticiones de cada uno dentro del mismo documento. Empleando la nomenclatura usada en la sección 3.3.1 podemos formular esta consideración de la siguiente forma:

$$w_{local}(t_j, d) = TF(t_j, d) \ , \tag{4.2}$$

donde $TF(t_j, d)$ identifica el número de apariciones del término t_j en el documento d, tal como vimos previamente en la ecuación 3.3.

Respecto al factor global, utilizaremos las frecuencias de término (TF_{ij}) y documento (DF_{ij}) teniendo en cuenta todas las muestras de entrenamiento de la categoría c_i y calculando el total de repeticiones en todas ellas. Queremos además que el valor de cada término t_j sea proporcional al número de caracteres (TLj) e inversamente proporcional a la frecuencia de categorías (CF_j). Podemos lograr esto aplicando el siguiente criterio:

$$w_{global}(t_j, c_i) = \frac{(TF_{ij} + DF_{ij}) \cdot TL_j}{CF_j} \ . \tag{4.3}$$

Como se puede observar a simple vista, mediante estas ecuaciones conseguimos que el peso sea proporcional a la longitud y las frecuencias del término y documento y forzamos a que sea menor cuando el término está más distribuido entre las diferentes categorías. De esta forma logramos que el esquema de ponderación propuesto contenga mucha más información que otros planteamientos más habituales. Por ejemplo, consideremos el caso del esquema TFIDF, donde el factor global (IDF) sólo proporciona una valoración para cada característica, asignando un único peso a cada término, sin tener en cuenta los matices que nos proporciona el contexto donde aparece.

Normalización

Los corpus de cada categoría pueden presentar diferencias considerables (por ejemplo, pueden existir variaciones en la cantidad de documentos de entrenamiento, los términos pueden contener más o menos palabras, etc.). Por tal motivo, no deberíamos usar los valores TF, DF y TL, tal y como los hemos obtenido al procesar el corpus. Si no aplicamos algún tipo de transformación, no podremos realizar comparaciones entre las distintas clases ya que, de lo contrario, aquellas con mayor número de muestras serían siempre las más probables. Una solución consiste en normalizar los valores de forma que los pesos finales no dependan del tamaño de los conjuntos de entrenamiento de cada categoría, ni de las diferencias de longitud de los términos representativos.

Del mismo modo que viene especificado en [Singhal et al., 1996], nosotros consideramos que los términos cuyas frecuencias y longitudes están muy cerca de la media son las más adecuados y sus pesos deberían permanecer igual, es decir, no deberían tener normalización.

Mediante una normalización media como el factor de pivote (w_{norm} en la ecuación 2.1), los valores normalizados para TF, DF y TL (TFn, DFn y TLn) se calculan obteniendo la proporción entre los valores totales y la media de todos los términos en la categoría. Los valores mayores que 1 indican una relevancia mayor que la media y, por lo tanto, indican la existencia de términos más relevantes:

$$TFn_{ij} = \frac{TF_{ij}}{\overline{TF_i}} \quad ; \quad DFn_{ij} = \frac{DF_{ij}}{\overline{DF_i}} \quad ; \quad TLn_{ij} = \frac{TL_j}{\overline{TL_i}} \ , \tag{4.4}$$

donde i apunta a la categoría c_i y j al término t_j.

Universitat d'Alacant
Universidad de Alicante

Descriptores de categorías

Respecto a las expresiones que coinciden con descriptores de categorías convendría hacer una mención especial. Siguiendo las intuiciones marcadas en el apartado 4.1.1 y utilizando el hecho de que los valores normalizados miden cuánto destaca un término sobre la media, podemos potenciar de manera sencilla el efecto de tales expresiones. Puesto que los descriptores de las categorías son expresiones especiales que pueden ser consideradas más importantes que la media, si un término t_j contiene alguno de los descriptores de c_i, sus frecuencias y longitud normalizadas deberían ser 1 o mayor que 1. Para asegurar esto, asignaremos a TFn, DFn y TLn valor 1 para todas las características que coincidan textualmente con algunos de los descriptores de categoría y que tengan pesos normalizados menores que 1.

4.3. Reducción de la dimensionalidad

Vamos a utilizar la frecuencia de categorías (CF) como medida de discriminación entre las distintas clases, combinándola con las frecuencias de término y documento, para aislar un conjunto de términos relevantes usando la media como el umbral de corte para decidir qué características deben representar cada categoría. También abordaremos mediante este criterio la automatización del proceso de selección de palabras de parada, ajustando la lista a cada corpus, tratando de encontrar en cada caso expresiones muy comunes dentro de la colección de entrenamiento que se esté utilizando.

4.3.1. Discriminación basada en la media

La media o tendencia central de un conjunto (lista) de valores se refiere a una medida del valor medio del conjunto de datos. En nuestro caso, si tenemos un conjunto T de n términos donde cada expresión se pondera conforme a una medida W ($\{w_1, w_2, \ldots w_n\}$), la media del conjunto T para la medida W (que denominamos $\overline{W}$) se define como la media aritmética para los valores w_i:

$$\overline{W} = \frac{\sum_{i=1}^{n} w_i}{n} \quad . \tag{4.5}$$

La discriminación media [Ruiz-Rico et al., 2009] utiliza la media de una medida W sobre un conjunto T como umbral de discriminación para descartar aquellos elementos de T cuyo peso w_i es mayor (*H-Average discrimination*) o menor (*L-Average discrimination*) que $\overline{W}$.

En el contexto de este trabajo, aplicaremos esta técnica sobre diferentes medidas para seleccionar los términos que tengan mayor capacidad discriminativa o aquellos que resulten más representativos de entre todos los que aparezcan en los documentos de entrenamiento.

4.3.2. Estrategias de reducción

Utilizando las medidas TF, DF y CF se puede reducir el número de dimensiones de los vectores de características mediante la definición de umbrales. Para cada categoría c_i proponemos aplicar tres filtros, los cuales enumeramos a continuación:

1. Seleccionar el conjunto de términos que son discriminativos entre categorías (*Category Discriminative Terms - CDT*).
2. Seleccionar el conjunto de términos que son predominantes dentro del corpus (*Best Terms in corpus - BT*).
3. Seleccionar el conjunto de términos que son predominantes dentro de cada categoría (*Best Terms in a Category - BTC*$_i$).

carcinoid (44,8)	*carcinoid heart disease (26,8)*
heart disease (40,9)	*tricuspid valve (11,5)*
carcinoid heart (26,8)	

Figura 4.2: Términos representativos de la categoría "Carcinoid Heart Disease", corpus OHSUMED, subárbol de enfermedades del corazón, una vez concluido el proceso de reducción de dimensionalidad.

Los términos más representativos de la categoría (*Most Representative Terms of the Category* - $MRTC_i$) se obtienen de la intersección de estos tres conjuntos de términos que acabamos de enumerar:

$$\{MRTC\}_i = \{CDT\} \cap \{BT\} \cap \{BTC\}_i \ . \tag{4.6}$$

Como resultado, se obtiene un subconjunto de términos para cada categoría. Por ejemplo, después de eliminar aquellos términos menos representativos, la categoría "Carcinoid Heart Disease" (corpus OHSUMED) se identifica por los términos que se muestran en la figura 4.2. Se puede comparar dicho ejemplo con el mostrado previamente en la figura 4.1, antes del proceso de reducción de la dimensionalidad, donde se incluían otros términos ahora descartados por su baja relevancia.

A continuación procederemos a establecer las bases que justifican y especifican el procedimiento de obtención de los conjuntos CDT, BT y BTC_i.

Discriminación entre categorías (CDT)

Algunos autores utilizan una lista fija de palabras de parada [Granitzer, 2003] para descartar términos durante el proceso de aprendizaje y clasificación. Sin embargo, esto puede no resultar adecuado para algunos corpus. El sistema propuesto aquí es capaz de generar esta lista sin ayuda de ningún experto, de forma que se ajusta automáticamente a las categorías, documentos y vocabulario de la colección de aprendizaje.

Partiendo de la afirmación de que el poder discriminativo de un término es inversamente proporcional al número de categorías que representa, podemos llegar a eliminar aquellos que resulten muy comunes. Nosotros sugerimos despreciar los términos que aparezcan en más de la mitad de las categorías por no considerarlos suficientemente discriminativos. Aunque no existe una justificación teórica que proporcione un umbral de corte óptimo, durante el proceso de evaluación sí observaremos evidencias empíricas de la mejora que proporciona el uso de la discriminación entre categorías basada en la media.

En este caso, el conjunto de los términos discriminativos para una categoría (CDT) se obtiene eliminando los términos que son representativos en más de la mitad de las clases. Esto es, un término t_j se considera palabra de parada si:

$$CF_j > (m/2 + 1) \ , \tag{4.7}$$

donde CF_j se puede calcular de la forma descrita en la ecuación 4.1 y m representa el número total de categorías.

Predominancia dentro de todo el corpus (BT)

Una vez eliminados aquellos términos considerados muy comunes por encontrarse distribuidos entre muchas categorías, la importancia de un término será proporcional al número de veces que aparezca dentro de todos los documentos de la colección de aprendizaje. Por ello, aplicaremos la discriminación L-Average (ecuación 4.5) utilizando los valores de TF y DF calculados sobre todos los términos en el corpus ($\overline{TF}$, $\overline{DF}$) para identificar y seleccionar los mejores términos (BT) en todo el corpus. En otras palabras, descartaremos aquellos términos t_j cuyas frecuencia de término y documento (TF_j y DF_j) estén por debajo de la media:

$$(TF_j < \overline{TF}) \wedge (DF_j < \overline{DF}) \ . \tag{4.8}$$

Predominancia en una categoría específica (BTC$_i$)

Finalmente, cuando hayamos descartado aquellos términos poco discriminativos por representar muchas categorías y también aquellos poco representativos por una baja aparición en el corpus, cuantas más veces aparezca un término en una categoría, más representativo será para esa categoría en particular. Para seleccionar los términos más adecuados en una categoría (BTC$_i$), se puede hacer uso nuevamente de la discriminación L-Average con los valores TF y DF calculados sobre todos los términos en cada categoría c_i. De esta forma, proponemos prescindir de aquellos términos t_j cuyas frecuencia de término y documento (TF_j y DF_j) tengan un valor inferior al promediado en la categoría c_i ($\overline{TF_i}$, $\overline{DF_i}$):

$$(TF_j < \overline{TF_i}) \wedge (DF_j < \overline{DF_i}) \ . \tag{4.9}$$

4.4. Algoritmo de clasificación

Para valorar las características seleccionadas hemos sugerido un cálculo de pesos conforme a una fórmula que nos permitirá obtener valores específicos de cada término para cada posible categoría. De esta forma, resulta sencillo conocer la similitud temática de un documento nuevo con respecto a cada clase, pues sólo necesitamos calcular la suma de todos los pesos de los términos contenidos en un texto para averiguar su grado de pertenencia a una clase en particular. En los siguiente párrafos vamos a describir los pasos que conforman el algoritmo de clasificación desarrollado en esta tesis.

4.4.1. Generación de vectores representativos de categorías

La existencia en el mundo real de enormes colecciones de documentos es una de las razones por las que la clasificación basada en un prototipo único por categoría es computacionalmente más atractiva que la metodología empleada por ejemplo en los algoritmos basados en los k vecinos más cercanos. Es obvio que en vez de comparar cada nuevo documento con cada vector que ya tenemos clasificado, resulta más rápido realizar comparaciones con un vector promediado de cada categoría. Las menores demandas computacionales de este tipo de algoritmos (basados en Rocchio) tienen un precio: la precisión obtenida no es tan buena como aquellos que utilizan para sus decisiones cálculos de similitud en tiempo real entre todas o gran parte de las muestras de entrenamiento [Bergo, 2002]. Si centramos nuestros esfuerzos en mejorar la extracción y ponderación de las características del corpus, probablemente conseguiremos incrementar la cantidad de información que nos permita identificar cada clase, aunque utilicemos para ello un solo vector. Logrando este objetivo, obtendremos sin duda clasificadores más precisos y, consecuentemente, conseguiremos tiempos muy reducidos tanto para el aprendizaje como para la clasificación sin tener que sacrificar por ello calidad en los resultados.

Nuestra aproximación, pues, se basa en la generación de vectores prototipo para cada categoría, que almacenen los datos suficientes para representar cada clase mediante un solo vector de características. Esto nos permitirá más adelante realizar comparaciones directas para poder obtener una lista ordenada de categorías calculando la similitud entre nuevos documentos y los vectores que hayamos obtenido después del proceso de entrenamiento.

Hay tres pasos principales implicados en la tarea de construir el vector representativo de cada categoría. Primero hemos analizado los datos de aprendizaje para detectar y extraer las expresiones relevantes (*selección de términos válidos*). En segundo lugar, hemos llevado a cabo un cálculo de pesos para cada una de las diferentes características que forman los vectores de categoría (*ponderación de términos*) utilizando una serie de estadísticas extraídas del corpus que contiene los documentos y los indicadores de pertenencia a las diferentes clases. Puesto que obtenemos un número elevado de características y éstas corresponden con cada una de las dimensiones de los vectores, hemos visto que resulta conveniente aplicar alguna técnica de *reducción de dimensionalidad*. Por ello, en último lugar hemos descartado ciertos términos que consideramos poco relevantes dentro de cada clase en particular o también poco discriminativos por su representatividad en la mayor parte de las clases.

Utilizando las características seleccionadas a continuación procederemos a construir un vector representativo por cada categoría, que intente abstraer en un sólo prototipo todos los documentos categorizados con la misma etiqueta.

Pesos en los prototipos de categorías

Puesto que pretendemos representar cada categoría mediante un único prototipo, no tiene sentido utilizar un factor local para su elaboración, ya que éste sólo contabiliza las apariciones

dentro de un documento en particular. Por ello, para generar los prototipos utilizaremos solamente el factor global después de aplicar la normalización a cada uno de sus componentes. Teniendo en cuenta que TF, DF y TL son indicativos de relevancia, incrementos en dichos valores deberán aumentar a su vez el peso del término, mientras que CF debería tener justo el comportamiento contrario. Partiendo de estas consideraciones, resulta sencillo combinar estas medidas en un solo valor que nos proporcione finalmente el peso de cada término t_j con respecto a cada categoría c_i. Podemos reducir dicho cálculo a la siguiente fórmula:

$$w_{ij} = w_{global}(t_j, c_i) = \frac{(TFn_{ij} + DFn_{ij}) \cdot TLn_{ij}}{CF_j} \ , \tag{4.10}$$

donde TFn_{ij}, DFn_{ij} y TLn_{ij} representan los valores normalizados de la frecuencia de término, documento y longitud del término t_j en la clase c_i, siendo CF_j el número de categorías en las que aparece dicho término.

Pesos en los vectores de nuevos documentos

Una vez entrenado el algoritmo,[5] podemos proporcionarle nuevos documentos con el propósito de obtener automáticamente una serie de etiquetas o categorías candidatas. Para ello, los nuevos textos deben ser procesados, al igual que se hizo con los documentos de entrenamiento. La diferencia radica en que el nuevo vector generado se utilizará para realizar comparaciones o evaluaciones de similitud con otros vectores para intentar establecer la pertenencia a unas u otras clases. Lógicamente, en este caso sí disponemos de la posibilidad de utilizar el factor de ponderación local, pues vamos a construir un vector de características a partir de un solo documento. En consecuencia podemos dejar reflejada una correspondencia directa entre el número de apariciones de los términos que contenga y cada uno de los pesos correspondientes:

$$\begin{aligned} w_{local}(t_j) &= TFnew_j \ , \\ w_{ij} = w_{local}(t_j) \cdot w_{global}(t_j, c_i) &= \frac{TFnew_j \cdot (TFn_{ij} + DFn_{ij}) \cdot TLn_{ij}}{CF_j} \ , \end{aligned} \tag{4.11}$$

donde $TFnew_j$ representa el número de veces que el término t_j aparece en el nuevo documento que está siendo representado como vector para ser clasificado. Conviene clarificar que éste el único valor que debe calcularse en tiempo real, pues se obtiene de un texto no visto anteriormente. El resto de parámetros de la ecuación pueden ser calculados off-line durante la fase de aprendizaje.

[5] Recordemos que en el caso de los clasificadores basados en prototipos, el entrenamiento concluye cuando se hayan generado los vectores representativos de cada categoría.

4.4.2. Clasificación mediante suma de pesos

Utilizando el método de obtención y ponderación de términos propuesto, no tenemos necesidad de emplear complejos algoritmos para clasificar nuevos textos. Si se representa un documento mediante un conjunto de características particularizado para cada clase, se puede calcular la relación entre este conjunto y las categorías a ordenar, obteniendo la suma de pesos de las características individuales coincidentes en el documento y en los vectores representativos de las categorías del corpus.

Utilizando los pesos w_{ij} obtenidos mediante la ecuación 4.11, cada documento D en el conjunto de evaluación se puede representar con una serie de vectores, formados por las características indicativas de pertenencia a cada categoría:

$$\begin{aligned} D &= \{\overrightarrow{d_1}, \overrightarrow{d_2}, \cdots, \overrightarrow{d_m}\} \\ c_1 \Rightarrow \overrightarrow{d_1} &= < w_{11}, \cdots, w_{1n} > \\ c_2 \Rightarrow \overrightarrow{d_2} &= < w_{21}, \cdots, w_{2n} > \\ &\cdots \\ c_m \Rightarrow \overrightarrow{d_m} &= < w_{m1}, \cdots, w_{mn} > \end{aligned}$$

donde m es el número de categorías y n es el número de términos representativos extraídos del documento D.

De esta forma, las categorías pueden ser ordenadas utilizando una suma de pesos (*Sum Of Weights - SOW*) como medida de similitud. Basta con realizar el siguiente cálculo para todos los términos t_j que aparecen tanto en el documento como en el vector de características representativo de la categoría c_i:

$$similitud(D, c_i) = \sum_{j=1}^{n} w_{ij} \ , \tag{4.12}$$

donde n es el número de términos válidos obtenidos al procesar el documento D, y w_{ij} el peso del término t_j para la categoría c_i.

Así, cuanto mayor sea el valor de la similitud, mayor será la relación entre el documento D y la clase c_i. Para obtener como resultado un ranking de categorías bastará con ordenar de forma descendente todas las categorías en una sola lista utilizando dicho valor, de forma que la más probable se mostrará en primer lugar.

4.4.3. Algoritmos no basados en prototipos

El método de ponderación propuesto puede ser utilizado también con clasificadores que no estén basados en prototipos, como SVM, aplicando la ecuación 4.11 para globalizar los

pesos, es decir, para calcular únicamente un peso por cada término independientemente de la categoría. Podríamos emplear cualquiera de las propuestas que se encuentran en la bibliografía (consultar por ejemplo las ecuaciones 3.6, 3.7 y 3.8). Sin embargo, como veremos a continuación, podemos particularizar los pesos para su uso en clasificadores binarios (volvemos a poner como ejemplo el algoritmo SVM), teniendo en cuenta que en tales casos los documentos utilizados para el aprendizaje se encuentran divididos en dos subconjuntos: muestras positivas y negativas.

Consideremos la clase c_p que contiene las muestras positivas en el conjunto de entrenamiento. Intuitivamente, para conseguir una representación que mantenga la distinción entre los elementos de dicho conjunto y el resto, los pesos de los términos que identifican c_p deberían calcularse de forma diferente de los que representan otras categorías.[6] Para ello debemos observar los dos subconjuntos de las muestras de entrenamiento, uno formado por aquellos documentos que se encuentran clasificados dentro de la categoría en cuestión, y el resto. Si el término aparece en los documentos del primer conjunto, su peso se puede calcular específicamente para la categoría a la que pertenece. Si el término no aparece entre el conjunto de muestras positivas, sugerimos acumular los pesos de las clases negativas, es decir, proponemos calcular la suma del peso del término para el resto de categorías. Más formalmente, partiendo de la ecuación 4.11, proponemos obtener el peso w_j del término t_j de la siguiente forma:

$$w_j = \begin{cases} w_{pj} & \text{si } t_j \in c_p \\ \sum_{i=1}^{m} w_{ij} \ \forall i \neq p & \text{en otro caso} \end{cases}, \tag{4.13}$$

donde w_{pj} se refiere al peso calculado a partir de las muestras positivas, y $\sum_{i=1}^{m} w_{ij} \ \forall i \neq p$ representa el peso calculado considerando las muestras negativas.

[6] Recordemos que el algoritmo SVM intenta encontrar el hiperplano que separa los ejemplos de entrenamiento positivos de los negativos. Si la representación vectorial agudiza dicha separación, conseguiremos maximizar el margen y reducir errores en la clasificación.

Capítulo 5

Evaluación

Para comprobar la validez de las teorías y procedimientos expuestos en el capítulo anterior, hemos realizado una serie de experimentos y comparaciones que nos permitirán valorar los resultados obtenidos utilizando como referencia otros trabajos publicados previamente. A lo largo del presente capítulo describiremos los procedimientos que hemos seguido para llevar a cabo todas las pruebas. En primer lugar hablaremos sobre los corpus que hemos elegido, destacando la diversidad en los tipos de documentos y temáticas empleadas, así como la cantidad de clases que finalmente pueden ser asignadas. En segundo lugar describiremos las diferentes medidas empleadas, entre las que podemos destacar, por ejemplo, la precisión y la cobertura, que nos permitirán evaluar tanto la cantidad de documentos clasificados correctamente como la calidad del listado del ranking de categorías. En tercer lugar, compararemos nuestra aproximación con trabajos previos. Por último, analizaremos la contribución individual de cada uno de los parámetros establecidos durante los procesos de aprendizaje y clasificación para justificar los ajustes y consideraciones propuestas.

Universidad de Alicante

En resumen, proponemos llevar a cabo la evaluación con corpus muy diferentes (Reuters y OHSUMED) y comparar los resultados empleando diversos algoritmos (SVM, kNN, etc.) que a su vez utilicen diferentes características de aprendizaje (BOW, bigramas, etc.). Esto nos permitirá no sólo realizar comparaciones, sino que además podremos observar las variaciones en los resultados al emplear diferentes tipos de documentos y métodos de entrenamiento y clasificación.

5.1. Corpus de trabajo

En su forma más simple, se puede llamar corpus a cualquier colección de documentos que contenga información textual. Sin embargo, para nuestros propósitos no podemos limitarnos a escoger cualquier conjunto de datos de forma arbitraria, ya que debemos buscar una serie de implicaciones que van más allá del simple análisis textual. Los datos no sólo nos deben ofrecer información sobre sí mismos, es decir, sobre lo que contienen, sino que deben representar una

sección amplia de una temática específica.

Intentando encontrar una definición de corpus que se adecue a nuestros objetivos, podemos mencionar, por ejemplo, la que aparece en [EAGLES, 1996]: "Un corpus es una colección de muestras de un lenguaje que se seleccionan y ordenan de acuerdo con criterios lingüísticos explícitos con el propósito de que puedan ser usadas como ejemplo". Pero dicha definición todavía resulta demasiado genérica y no incluye ciertos requerimientos recomendables para evaluar correctamente un sistema de clasificación de textos. Podemos ampliarla concretando algunas de las condiciones que deberían cumplir los corpus empleados:

- La cantidad de datos de entrenamiento debe poder ser procesada por la tecnología computacional utilizada.
- Se deberían utilizar ejemplos que incluyan un amplio rango de información con el propósito de ofrecer suficiente representatividad.
- Los documentos deberían estar correctamente clasificados en temáticas específicas, o en subtemáticas bien diferenciadas dentro del mismo ámbito.
- El tamaño de los documentos debería ser uniforme.
- Los textos deberían carecer de restricciones que impidan su libre uso y distribución.

Aunque la mayoría de los corpus existentes en la actualidad no se ajustan a alguna de estas recomendaciones, no por ello han sido considerados inadecuados. Por ejemplo, refiriéndonos a los dos primeros puntos, es difícil encontrar el balance adecuado entre calidad y cantidad. Por un lado, nos encontramos con autores que dan mucha importancia al hecho de que el corpus sea representativo y equilibrado [Curran y Moens, 2002; Gauch et al., 2006] y por otro lado nos encontramos aquellos que destacan la importancia de que el corpus sea lo más cuantioso posible [Brückner et al., 2006; Pomikálek et al., 2009].

Resumiendo, un corpus de prueba idóneo para clasificación automática se puede definir como una colección de datos que incluye gran cantidad de documentos de tamaño uniforme, y que versan sobre temas concretos. Se deben proporcionar también las asignaciones de dichos documentos a diversas categorías, las cuales se encuentran perfectamente definidas y diferenciadas. Además, para poder evaluar de forma objetiva el comportamiento de un clasificador es imprescindible que dicha colección de pruebas haya sido utilizada ampliamente por otros investigadores con anterioridad, por lo que es recomendable utilizar textos cuyas licencias permitan su libre uso.

En adelante, para comprobar que nuestro algoritmo de clasificación funciona correctamente con distintos tipos de documentos y clases, utilizaremos dos corpus muy diferentes, bien conocidos y referenciados con frecuencia durante los últimos años: Reuters-21578 y OHSUMED, divididos a su vez en distintos subconjuntos.[1] De esta forma conseguiremos demostrar

[1] Se puede consultar el apéndice A para observar los descriptores de las categorías de cada conjunto.

que, a pesar de no utilizar colecciones de datos idóneas (algunas de las que emplearemos no cumplen con las recomendaciones mencionadas en los párrafos anteriores), el sistema consigue adaptarse automáticamente y sigue ofreciendo resultados coherentes aún cuando se introducen variaciones considerables en las muestras de entrenamiento.

5.1.1. Reuters-21578

Esta es probablemente la colección de pruebas que más se ha utilizado para tareas de clasificación automática [Debole y Sebastiani, 2004a]. Se encuentra formada por noticias muy cortas, distribuidas en 135 categorías, que se encuentran relacionadas fundamentalmente con el área de los negocios y la economía. Es una colección de tamaño relativamente pequeño, pues sólo contiene 21.578 documentos.[2] Además no todos ellos son válidos, pues algunos ni siquiera están categorizados. Intentando aprovechar el mayor número de muestras, sólo se pueden emplear como mucho 9.603 documentos para llevar a cabo el entrenamiento y 3.299 para comprobar la calidad de los resultados (estos conjuntos de entrenamiento y evaluación ya se encuentran predefinidos).

Podemos destacar las siguientes características que hacen que Reuters-21578 resulte una colección desafiante para la realización de experimentos de clasificación de textos:

- Permite pruebas de multietiquetado (cada documento puede pertenecer a más de una categoría).
- El conjunto de categorías no es exhaustivo (algunos documentos no pertenecen a ninguna categoría).
- La distribución de los documentos entre las diferentes categorías es muy desigual. Algunas categorías tienen muy pocas muestras positivas mientras que otras tienen miles de ellas.
- Existen ciertas relaciones semánticas entre categorías. Por ejemplo, existe una categoría denominada "trigo" ("wheat"), y otra denominada "grano" ("grain"), que obviamente están relacionas, pero de forma oculta (no se dispone de una organización jerárquica explícita definida para estas categorías).

Este corpus también resulta especialmente desafiante para sistemas de clasificación basados en técnicas de aprendizaje automático, puesto que varias categorías tienen muy pocos ejemplos de entrenamiento (sea cual sea la división entre documentos de entrenamiento y prueba) y además, se trata de documentos muy pequeños (contienen una media de sólo 45 palabras no repetidas [Schapire et al., 1998]). Esto hace que la construcción inductiva del clasificador no sea una tarea fácil.

[2] La colección completa se puede obtener desde http://www.daviddlewis.com/resources/testcollections/reuters21578/index.html.

Desafortunadamente, los beneficios que esta colección ha aportado a la investigación en la clasificación de textos han estado limitados por el hecho de que diferentes autores han construido diversas subcolecciones a partir de ésta, probando sus sistemas sólo con el nuevo subconjunto. La forma más frecuente en la que han efectuado estas subdivisiones ha sido mediante restricciones para seleccionar sólo un pequeño número de categorías. Algunos de los subconjuntos que se han mencionado con más frecuencia y que nosotros utilizaremos como referencia en las siguientes secciones son:

- El conjunto de las 10 categorías con mayor número de muestras positivas para el entrenamiento. Nosotros utilizaremos este subconjunto para realizar comparaciones con [Dumais et al., 1998; Antonie, 2002; Tesar et al., 2006]. En el gráfico 5.1 se pueden observar los nombres de dichas categorías junto con la cantidad de documentos de cada una, tanto para la fase de entrenamiento como para la posterior fase de evaluación.
- El conjunto de las 50 categorías con mayor número de muestras positivas para el entrenamiento. En este caso referenciaremos [Cai y Hofmann, 2003] para comparar resultados.
- El conjunto de las 90 categorías con al menos una muestra positiva tanto en el conjunto de entrenamiento como en el de prueba, que tomaremos como referencia para evaluar con respecto a [Joachims, 1998].

Respecto a las licencias de uso, "Reuters Ltd. and Carnegie Group, Inc" sigue siendo la empresa propietaria de esta colección de datos, pero permite la distribución gratuita para su uso dentro del ámbito de la investigación.

Universitat d'Alacant
Universidad de Alicante

5.1.2. OHSUMED

Esta colección está formada por 348.566 citas de revistas médicas publicadas entre 1987 y 1991. Cada documento se encuentra formado por un título y un resumen. Sin embargo, en algunos casos sólo se dispone del título, lo que nos deja con 233.445 documentos válidos.[3] Todos ellos se asociaron manualmente a una o varias categorías elegidas entre un total de 14.321 clases formadas por términos MeSH,[4] que pueden consistir en nombres de enfermedades, principios activos de medicamentos, tratamientos, etc. Tales términos se encuentran estructurados en una jerarquía donde los términos genéricos se encuentran cercanos al nodo raíz, y los más específicos se localizan a medida que se desciende en el árbol. Cada término aparece por lo menos en una posición de la jerarquía y además es frecuente que un término aparezca en diferentes posiciones.

El volumen de documentos proporcionados y el número elevado de posibles categorías derivan en un problema de clasificación bastante complejo. Citando por ejemplo [Yang, 1996], observamos que se propone utilizar todas las publicaciones entre 1987 y 1990 para realizar

[3] La colección completa puede descargarse desde http://ir.ohsu.edu/ohsumed/.
[4] Medical Subject Headings. Más información en http://www.nlm.nih.gov/mesh/meshhome.html.

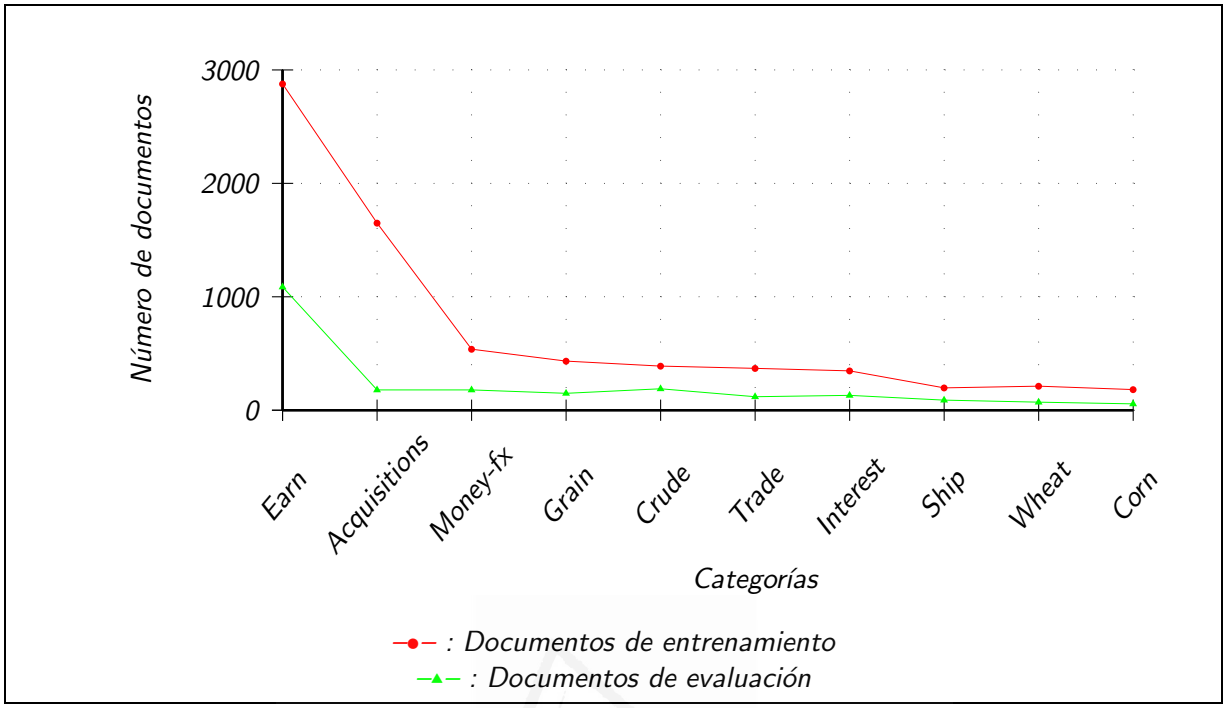


Figura 5.1: Número de documentos de entrenamiento y evaluación para las 10 categorías más pobladas de la colección Reuters-21578 [Manning et al., 2008].

el entrenamiento, lo que representa que el algoritmo de clasificación debe procesar 183.229 documentos y aprender las categorías asignadas a dichos documentos, a partir de un total de 24.939 características (palabras) únicas (en [Yang y Pedersen, 1997] se habla incluso de 76.076 características). Además, hablamos de textos que contienen una media de 167 palabras entre título y resumen, y que se encuentran etiquetados con una media de 12 categorías cada uno. Por ello existen muy pocos trabajos que hayan experimentado con el rango completo de categorías. La mayoría de los autores realizan divisiones del conjunto original y establecen una cantidad menor de documentos de entrenamiento y prueba. Por ejemplo, en [Rafi y Shaikh, 2013] se utilizan sólo 10 categorías y 800 documentos, o en [Borodin et al., 2013] podemos encontrar que también utilizan 10 categorías, aunque en este último caso se seleccionan 9.310 documentos. Nosotros hemos decidido emplear algunas agrupaciones que cuentan con numerosas referencias bibliográficas, lo que nos permitirá comparar mejor nuestros resultados y establecer conclusiones. Los tres subconjuntos de evaluación que proponemos vienen determinados por los sub-árboles de enfermedades y enfermedades del corazón, y por las 50 categorías con mayor número de muestras:

- Tal como se describe en [Joachims, 1998], para la jerarquía de las enfermedades, agrupamos las categorías MeSH por debajo del mismo nodo raíz, de forma que se obtienen 23 categorías. Los primeros 10.000 documentos en 1991 que contienen resúmenes se utilizan para el entrenamiento y los segundos 10.000 se usan para la evaluación.

- Para el sub-árbol de las enfermedades del corazón [Ruiz y Srinivasan, 2002; Granitzer,

Nivel de jerarquía	Categorías	Documentos de entrenamiento	Documentos de evaluación
1	1	673	177
2	23	10.269	2.987
3	56	6.141	1.732
4	16	1.164	329
5	6	159	38

Tabla 5.1: Número de categorías y cantidad de documentos en el conjunto de entrenamiento y evaluación en los 5 primeros niveles del corpus OHSUMED, sub-árbol de enfermedades del corazón.

2003], se descartan las categorías que no contienen documentos de aprendizaje, quedando sólo 16.592 documentos y 102 posibles categorías. En este último caso, los documentos de 1987 a 1990 se usan para el aprendizaje y los de 1991 se usan para las pruebas. En la tabla 5.1 podemos observar la cantidad de categorías y documentos en los 5 primeros niveles de esta rama.

- Con el objetivo de disponer de un conjunto de entrenamiento robusto y homogéneo en el número de documentos de entrenamiento y evaluación, también vamos a utilizar el subconjunto de categorías compuesto por las 50 primeras clases que contienen mayor número de documentos asociados. En este caso dispondremos de un total de 54.708 documentos, de los cuales seleccionaremos el 90 % para la fase de entrenamiento y el 10 % para la evaluación, siguiendo el mismo criterio que el empleado en [Cai y Hofmann, 2003].

Universitat d'Alacant
Universidad de Alicante

La tarea de clasificación con OHSUMED se considera más ardua que con Reuters, tal como se observa en trabajos previos [Yang, 1996]. La causa radica principalmente en que el número de categorías por documento es mucho mayor, pues nos encontramos con una media de 12 categorías por documento, frente a 1,2 en Reuters. Esto repercute en una correspondencia mucho más difusa entre los conjuntos de términos representativos y las relaciones de pertenencia a las categorías.

5.2. Medidas de evaluación

Para la mayoría de los investigadores ha resultado difícil comparar las diversas técnicas utilizadas durante el proceso de clasificación, debido a que no se cuenta con una medida única de evaluación ni se dispone de un conjunto de datos universal y común para todos. En muchos experimentos las colecciones de datos y las evaluaciones se han adaptado de acuerdo con los algoritmos que se pretendía probar. Además, es importante aclarar que la mayoría de trabajos en el área pretenden determinar principalmente la efectividad de los clasificadores, sin tener en cuenta su eficiencia.

De entre las medidas estándar usadas más frecuentemente, hemos elegido las siguientes para realizar comparaciones y evaluar el funcionamiento del algoritmo propuesto: precisión, cobertura, F1 y BEP [Yang, 1999; Joachims, 2000]. Exponemos a continuación el procedimiento para su obtención, así como la descripción de otras medidas, que aunque no mencionadas tan activamente en la bibliografía, también han sido consideradas en esta evaluación.

5.2.1. Precisión y cobertura

La precisión es el porcentaje de asignaciones correctas del sistema, es decir, la relación entre las asignaciones correctas y el número total de asignaciones del sistema. La cobertura se define como la relación entre asignaciones correctas del sistema y el número total de asignaciones correctas. En otras palabras, la precisión expresa en qué medida el clasificador toma una decisión correcta al ubicar cualquier documento en la clase que le corresponde, y la cobertura refleja cuántos de todos los documentos de una clase son clasificados en ella.

Empleando la nomenclatura utilizada en [Yang, 1999], podemos calcular la precisión y la cobertura de la siguiente forma:

$$Precisión = \frac{Categorías\ asignadas\ correctamente}{Total\ de\ categorías\ asignadas} \quad , \tag{5.1}$$

$$Cobertura = \frac{Categorías\ asignadas\ correctamente}{Total\ de\ categorías\ correctas} \quad . \tag{5.2}$$

Universitat d'Alacant
Universidad de Alicante

En [Sebastiani, 2002] encontramos, sin embargo, una definición más completa y ampliamente referenciada en la bibliografía. Procedemos a detallarla a continuación y mantendremos dicha nomenclatura en definiciones posteriores.

Haciendo referencia a la definición de clasificación de textos descrita anteriormente (sección 2.1.3), partiremos de un conjunto documentos $D = \{d_1, ..., d_m\}$ y de un conjunto de categorías $C = \{c_1, \ldots, c_n\}$. De esta forma, un clasificador intentará aproximar la función $D \times C \rightarrow \{V, F\}$, donde V y F representan "Verdadero" y "Falso" respectivamente. Para intentar medir la calidad del proceso de clasificación, se plantea la siguiente matriz de contingencia para cada una de las categorías c_i:

Categoría c_i		Decisión del experto humano	
		SÍ	NO
Decisiones del clasificador	SÍ	VP_i	FP_i
	NO	FN_i	VN_i

donde:

- VP_i (Verdaderos Positivos) es el número de documentos que pertenecen a c_i y que han sido clasificados correctamente.
- VN_i (Verdaderos Negativos) es el número de documentos que no pertenecen a c_i y que han sido clasificados correctamente.
- FP_i (Falsos Positivos) es el número de documentos que no pertenecen a c_i y que han sido clasificados de manera incorrecta.
- FN_i (Falsos Negativos) es el número de documentos que pertenecen a c_i y que han sido clasificados de manera incorrecta.

Y para unificar dichos valores se aplican las siguientes fórmulas:

$$Precisión_i = \frac{VP_i}{VP_i + FP_i} \ , \tag{5.3}$$

$$Cobertura_i = \frac{VP_i}{VP_i + FN_i} \ . \tag{5.4}$$

La precisión máxima es 1 por definición y supone que todas las categorías asignadas son correctas. La cobertura máxima es también 1 por definición, y conlleva la clasificación de todos los documentos relevantes existentes.

Ambas medidas son complementarias y siempre suelen evaluarse juntas, puesto que por separado pierden su validez. Por ejemplo, si se devuelve un único documento que resulta ser correcto ($TP = 1, FP = 0$), la precisión sería 1 ya que no habría ningún resultado irrelevante (sin embargo, la cobertura podría ser muy baja). Por otro lado, se puede garantizar trivialmente una cobertura total asignando una categoría a todos los documentos de la colección para cualquier consulta (en este caso, la precisión sería muy baja).

Se definen a su vez *Precisión(r)* y *Cobertura(r)* como los valores de precisión y cobertura calculados a partir de las categorías situadas por encima de un determinado umbral r en una lista ordenada o ranking de categorías. En otras palabras, la Precisión(r) refleja la proporción entre el número de categorías relevantes en las r primeras posiciones y la cantidad de categorías mostradas (r). La Cobertura(r) viene definida por la proporción entre la cantidad de categorías correctas encontradas hasta la posición r y la cantidad total de categorías relevantes para un documento.

Por ejemplo, supongamos que para un documento se obtienen 10 categorías de las cuales 4 son relevantes de un total de 7. En este caso la Precisión(10) es de $4/10 = 0,4$ y la Cobertura(10) es $4/7 = 0,57$. Si se obtienen 10 categorías más y ninguna de ellas es relevante entonces la Precisión(20) será de $4/20 = 0,2$ y la Cobertura(20) será la misma que en el caso anterior $(0,57)$. Dicho de otro modo:

$$Precisión(r) = \frac{\sum_{i=1}^{r} rel(i)}{r} \ , \tag{5.5}$$

$$Cobertura(r) = \frac{\sum_{i=1}^{r} rel(i)}{Total\ de\ categorías\ correctas} \ , \tag{5.6}$$

donde r es el umbral de resultados que tenemos en cuenta, y $rel(i)$ retorna 1 si el resultado i-ésimo es relevante para la consulta y 0 en caso contrario.

También nos podemos encontrar con una variante de la *Cobertura(r)*. En [Crammer y Singer, 2002; Amit et al., 2007; Bhowmick et al., 2010] se habla de *Coverage* (*Cov*), cuyo valor se utiliza para medir cuánto tenemos que descender en la lista de categorías para cubrir todos los resultados relevantes. Partiendo de un documento en particular, se genera una lista de categorías ordenadas por relevancia, y se coge el número máximo de etiquetas que incluya todas las categorías correctas a las que pertenece tal documento. Después se divide dicho valor entre el número total de categorías correctas, y se le resta la unidad para que la cobertura tenga valor cero cuando se obtenga un ranking perfecto:

$$Cov = \frac{max\ r}{Total\ de\ categorías\ correctas} - 1 \ . \tag{5.7}$$

5.2.2. Punto de equilibrio entre precisión y cobertura (BEP)

Debemos recordar que la precisión y la cobertura tomadas por separado presentan el problema de que se puede aumentar una disminuyendo la otra y viceversa. Para evitar dicho efecto, ambas medidas pueden combinarse en un solo valor con el propósito de evaluar un balance entre ambas para reflejar mejor el equilibrio entre la cantidad y la calidad de los resultados mostrados como relevantes. Existen diversas propuestas, entre las que podemos encontrar BEP (*Precisión/Recall Break-Even Point*) [Lewis, 1992a]. Es el valor que se obtiene cuando la precisión y la cobertura son iguales, es decir, cuando el número de ejemplos de prueba que se clasifiquen en la clase positiva sea igual al número real de ejemplos de prueba en la clase positiva ($VP + FP = VP + FN$).

Se calcula obteniendo los valores de $Precisión(r)$ y $Cobertura(r)$ tomando *r* como el máximo número de categorías relevantes para cada documento:

$$r = VP + FN \rightarrow BEP = Precisión(r) = Cobertura(r) \ . \tag{5.8}$$

5.2.3. F1

Aunque BEP nos proporciona una medida de evaluación de un solo valor, presenta como inconveniente que puede favorecer resultados triviales. Por ejemplo, la precisión será muy

alta clasificando un documento como relevante en muy pocas categorías, y podemos obtener de este modo un BEP próximo a $0,5$, ya que nos basamos en una media no ponderada [Bekkerman et al., 2001]. Para evitar estas situaciones se suele emplear la medida F, que se ha convertido en una de las más populares. Se utilizó por primera vez en [Van Rijsbergen, 1979], donde fue propuesta como una medida única para evaluar la validez de un sistema de recuperación de información. Se define como sigue:

$$F = \frac{1}{\alpha \frac{1}{Precisión} + (1-\alpha)\frac{1}{Cobertura}} \ . \tag{5.9}$$

Se suele utilizar un valor de $\alpha = \frac{1}{2}$, por lo que la ecuación generalmente empleada para calcular la medida F (también conocida como F1) es la siguiente:

$$F_1 = \frac{2 \cdot Precisión \cdot Cobertura}{Precisión + Cobertura} \ . \tag{5.10}$$

Mediante este cálculo, si la precisión es muy alta a costa de obtener muy poca cobertura, el valor de F1 será también muy bajo (próximo a cero), pues estamos hablando de una media armónica [Bekkerman et al., 2001].

Una variante que también suele aparecer en la bibliografía es el valor de F1 máximo (*maxF1*), que se define como el valor máximo calculado a partir de los *r* valores que *F1* puede tomar. Partiendo de la Precisión y la Cobertura en *r*, podemos calcular *F1(r)* y por lo tanto *maxF1* de la siguiente forma:

$$F_1(r) = \frac{2 \cdot Precisión(r) \cdot Cobertura(r)}{Precisión(r) + Cobertura(r)}, \tag{5.11}$$

$$maxF1 = \max_r F1(r) \ . \tag{5.12}$$

5.2.4. Precisión media (AvgP)

Mide la proporción de categorías relevantes en una lista ordenada. La Precisión en *r* se calcula para cada categoría relevante, y la media resultante se calcula a partir de todas las posiciones de las categorías relevantes en la lista ordenada. Una clasificación perfecta implica AvgP con valor 1.

Por ejemplo, dado el siguiente conjunto de etiquetas o categorías:

$$\{A, B, C, D, E, F, G, H, I, J, K, L, M, N, O\}$$

supongamos que a un documento dado, se le asignan la siguiente lista de categorías:

$$\{\underline{E}, L, M, \underline{I}, \underline{O}, C, N, D, F, \underline{A}, H, G, B, J, K\}$$

donde se muestran subrayadas las categorías relevantes para ese documento. Para calcular la precisión media es necesario calcular la precisión cada vez que se asigna una categoría al documento hasta haber completado todas los categorías relevantes. Así, los valores de precisión serían: 1 al asignar la categoría E; 0,5 al asignar la categoría I; 0,6 al asignar la categoría O y 0,4 al asignar la categoría A. Por tanto la precisión media sería:

$$AvgP = \frac{1 + 0,5 + 0,6 + 0,4}{4} = \frac{2,5}{4} = 0,625 \ .$$

Dicho de otro modo:

$$AvgP = \frac{\sum_{r=1}^{N} Precisión(r) \cdot rel(r)}{R} \ , \tag{5.13}$$

donde:

- N es el número de resultados.
- r es el resultado r-ésimo.
- $Precisión(r)$ es la precisión en r.
- $rel(r)$ es 1 si el resultado r-ésimo es relevante para la consulta y 0 en caso contrario.
- R es el número de categorías relevantes para la consulta realizada sobre un documento dado.

Universitat d'Alacant
Universidad de Alicante

5.2.5. One-Error (OneErr) e Is-Error (IsErr)

La medida $OneErr$ toma valor 1 si la categoría obtenida en primera posición para un documento en particular no es una de las categorías relevantes. Se define como 0 en caso contrario:

$$OneErr = \begin{cases} 1 & \text{si } rel(1) \neq 1 \\ 0 & \text{en otro caso} \end{cases} \ . \tag{5.14}$$

De este modo resulta útil calcular la media de $OneErr$ sobre un corpus, pues se consigue reflejar la fracción de veces que la respuesta indicada en primera posición no es una categoría relevante.

	Micropromedio	Macropromedio
Precisión	$\frac{\sum_{i=1}^{\|C\|} VP_i}{\sum_{i=1}^{\|C\|} VP_i + FP_i}$	$\frac{\sum_{i=1}^{\|C\|} \frac{VP_i}{VP_i + FP_i}}{\|C\|}$
Cobertura	$\frac{\sum_{i=1}^{\|C\|} VP_i}{\sum_{i=1}^{\|C\|} VP_i + FN_i}$	$\frac{\sum_{i=1}^{\|C\|} \frac{VP_i}{VP_i + FN_i}}{\|C\|}$

Tabla 5.2: Cálculo de la precisión y cobertura micro y macro promediadas.

Is-Error ($IsErr$) es simplemente el indicador de si el ranking obtenido es perfecto o no. $IsErr$ es 1 si hay alguna categoría errónea, y 0 en cualquier otro caso:

$$IsErr = \begin{cases} 1 & \text{si } \sum_{i=1}^{N} rel(i) \neq N \\ 0 & \text{en otro caso} \end{cases} . \tag{5.15}$$

5.2.6. Micropromedio y macropromedio

Existen dos formas de promediar los resultados obtenidos al evaluar sistemas de categorización automática [Lewis, 1991]. Dado un conjunto de documentos D y una serie de C categorías, un categorizador toma $D \cdot C$ decisiones que pueden ser evaluadas individualmente. Para ofrecer al final un único valor pueden considerarse todas las decisiones como pertenecientes a un solo conjunto, y como tal calcular un valor promedio directamente. Por otro lado, también se puede obtener la precisión y la cobertura para cada categoría y posteriormente calcular la media entre todas las categorías. En el primer caso se habla de micropromedio (*microaveraging*) y en el segundo de macropromedio (*macroaveraging*), tal como se refleja en la tabla 5.2.

La diferencia entre una y otra medida es simple: en el caso del micropromedio tiene más influencia el resultado global (esto es, el número total de categorizaciones correctas) frente a las diferencias entre categorías (puede haber diferencias notables entre los resultados obtenidos para cada categoría), mientras que en el caso del macropromedio influyen más las diferencias entre categorías que los resultados tomados en su conjunto, es decir, se premiaría al categorizador que obtiene resultados similares en todas las categorías. De esta forma, el macropromedio permite evaluar la precisión del clasificador con independencia del número de muestras por clase. Debemos tener esto en cuenta, pues podríamos evaluar erróneamente un clasificador. Por ejemplo, si tenemos varias clases con muchas más muestras que en el

Categoría	Resultados	Precisión
EN	950 (EN) - 50 (DE)	0,95
ES	600 (ES) - 100 (PT)	0,86
PT	150 (PT) - 10 (FR)	0,94
DE	50 (DE) - 50 (EN)	0,50
FR	10 (FR) - 50 (PT)	0,17
Promedio	$Micro = \frac{1760}{2020} = 0,87$	$Macro = \frac{0,95 + 0,86 + 0,94 + 0,5 + 0,17}{5} = 0,68$

Tabla 5.3: Ejemplo de cálculo de precisión mediante micro y macro promedios.

resto, la precisión en dichas clases tenderá a ser muy alta, y elevará por tanto el promedio del clasificador, aunque hayan categorías con precisión casi nula.

A continuación mostramos un pequeño ejemplo para ilustrar las diferencias en los cálculos entre uno y otro promedio. Supongamos que disponemos de una colección de documentos en distintos idiomas: 1000 en inglés (EN), 600 en español (ES), 300 en portugués (PT), 100 en alemán (DE) y 20 en francés (FR). Tomando como categorías los nombres de los idiomas, el objetivo será categorizar los documentos de acuerdo a su idioma, teniendo en cuenta que no conocemos a priori la lengua en la que están escritos. Por último supongamos que los resultados obtenidos al categorizar automáticamente dicha colección son los obtenidos en la tabla 5.3. Observamos que el valor macropromedio es de 0,68 y el micropromedio de 0,87. Es decir, un 87 % del total de documentos fueron clasificados correctamente; sin embargo, puesto que el valor macropromediado es mucho menor puede afirmarse que este sistema se comporta de manera sustancialmente diferente frente a las distintas categorías.

Universitat d'Alacant
Universidad de Alicante

5.3. Experimentos

A continuación vamos a describir los experimentos llevados a cabo para evaluar el sistema de clasificación propuesto.

En primer lugar evaluaremos sólo el primer proceso llevado a cabo por el clasificador: la selección de características. Emplearemos clasificadores estándar, que utilizarán los vectores de características generados por nuestro algoritmo para completar su entrenamiento, y así poder realizar comparaciones de un mismo sistema de clasificación empleando unos u otros conjuntos de características.

En segundo lugar le proporcionaremos al clasificador nuevos documentos para que nos devuelva un listado de categorías propuestas y podamos evaluar la calidad de los resultados obtenidos por un usuario final que utilice el sistema completo.

Por último, intentaremos aislar cada uno de los factores considerados en nuestro algoritmo

para observar la influencia de cada uno sobre el resultado final.

5.3.1. Selección de características

A continuación intentaremos valorar la calidad de las expresiones representativas seleccionadas junto con los pesos asignados. Para ello, construiremos los vectores de características aplicando nuestra aproximación a todos los documentos de prueba, y a continuación, entrenaremos y evaluaremos el rendimiento del algoritmo SVM con tales datos. Esto nos permitirá comparar los valores obtenidos con los que podemos encontrar en la bibliografía y que utilizan el mismo algoritmo de clasificación, pero emplean otros métodos para la elaboración de los vectores de características (BOW, bigramas, etc.).

Respecto a la transformación de los datos de entrada, seguiremos las pautas marcadas en el capítulo anterior: primero llevaremos a cabo un proceso de obtención de características tal como expusimos en la sección 4.1.3 (selección de términos válidos), y a continuación asignaremos una valoración a cada una de las expresiones utilizando los cálculos descritos en la sección 4.2.5 (ponderación de términos). Más concretamente, utilizaremos la ecuación 4.13 para obtener finalmente los pesos que emplearemos para alimentar el algoritmo SVM.

Tal como se propone en [Dumais et al., 1998; Antonie, 2002; Granitzer, 2003], utilizaremos el software SVM^{light} [Joachims, 1999] como clasificador base para realizar la evaluación. Para conseguir el comportamiento óptimo en cada categoría hemos ajustado algunos parámetros y hemos tenido en cuenta ciertas consideraciones, tal como enumeramos a continuación [Joachims, 2002, 2003]:

- **Normalización**. Puesto que SVM ofrece mejores cotas de error usando la norma euclídea [Zu et al., 2003; Granitzer, 2003], hemos normalizado todos los vectores de características (tanto en el conjunto de entrenamiento como de pruebas) para obtener una longitud euclídea unitaria.

- **Kernel**. Hemos utilizado el kernel lineal con los ajustes óptimos proporcionados por la herramienta *easy.py*,[5] incluida dentro de la librería LIBSVM [Chang y Lin, 2001], desarrollada para facilitar el uso del clasificador SVM, ajustando los parámetros adecuados para reproducir experimentos y comparar resultados, o para optimizar los ajustes dependiendo del corpus. Respecto a los valores de γ, para realizar comparaciones con [Joachims, 1998], los hemos ajustado manualmente para adecuarlos a los establecidos en dicha publicación. Este parámetro controla la forma de separar los vectores de soporte. Generalmente, incrementos de γ ocasionan el uso de un mayor número de vectores para llegar a construir el clasificador.

- **Parámetro C**. Sirve para ampliar el margen de error en la clasificación de las muestras de entrenamiento, consiguiendo de esta forma generalizar el modelo vectorial, y mejorando en consecuencia los resultados al clasificar nuevos documentos que tal vez no se

[5] http://www.csie.ntu.edu.tw/~cjlin/libsvm/

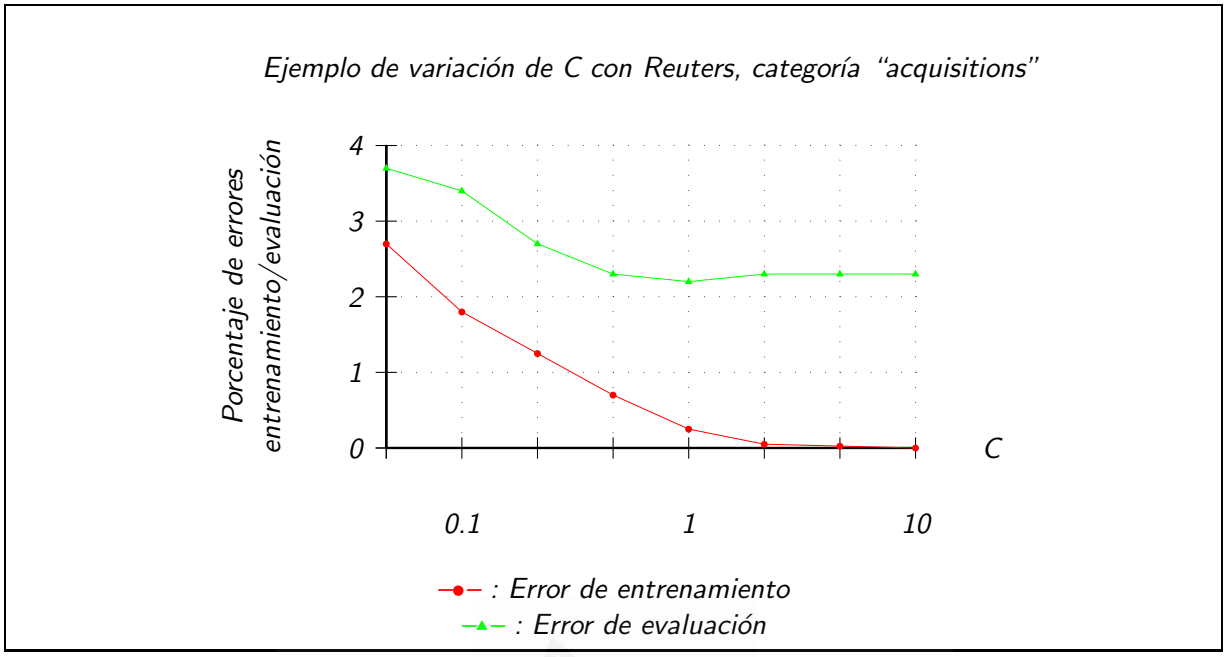


Figura 5.2: Ejemplo de comportamiento para SVM variando el parámetro C. Se han utilizado los documentos del corpus Reuters en la categoría "acquisitions".

ajustan perfectamente a ninguno de los vectores de soporte escogidos por el algoritmo inicialmente. La figura 5.2 muestra un ejemplo del comportamiento de SVM para diferentes valores de C, cuyo valor se puede introducir mediante el parámetro "-c" desde la línea de comandos. En nuestro caso, hemos limitado los ajustes a 5 posibles valores enteros, comprendidos entre 1 y 5.

- **Factor de coste**. Se utiliza para controlar el peso de los ejemplos positivos en relación con los negativos, y por lo tanto, nos proporciona una forma de compensar las clases desequilibradas [Morik et al., 1999]. Se puede ajustar mediante el parámetro "-j" desde la línea de comandos. Hemos escogido el valor más adecuado obtenido de manera empírica de entre 16 posibles valores comprendidos entre $0,25$ y 4 considerando incrementos de $0,25$: $0,25$, $0,5$, $0,75 \cdots 3,75$, $4,0$.

- **Validación cruzada**. Para calcular el valor más adecuado para cada categoría del parámetro C y el factor de coste, utilizamos la validación cruzada "Leave-one-out" (LOO) (activada mediante el parámetro "-x 1") que nos permite obtener la precisión resultante para cada ajuste utilizando sólo el conjunto de entrenamiento para generar automáticamente diversos subconjuntos de aprendizaje y prueba [Lau y Wu, 2004].

Análisis de la selección y ponderación de características

En las tablas 5.4, 5.5, 5.6 y 5.7 podemos observar diferentes medidas obtenidas utilizando el algoritmo de clasificación SVM entrenado y evaluado mediante vectores de características

	OneErr	IsErr	AvgP	BEP	maxF1	Cov
SVM+BOW+TFIDF	4,09	4,72	97,62	93,69	98,28	0,17
SVM+BOW+Pond. NEWPAR	4,05	4,87	97,61	93,51	98,25	0,17
SVM+BOW (Sust.)+Pond. NEWPAR	3,69	**4,28**	97,79	94,09	98,37	0,17
SVM+N-GRAMAS+Pond. NEWPAR	**3,46**	4,64	**97,86**	**94,14**	**98,37**	**0,17**

Tabla 5.4: Valoraciones micropromediadas al utilizar diferentes características y ponderaciones sobre el corpus Reuters, 10 categorías con mayor número de muestras.

construidos mediante nuestro algoritmo. Hemos utilizado la discriminación basada en la media (*H-Average discrimination*) para la selección automática de las palabras de parada de cada corpus, de forma que sean las expresiones más frecuentes las que seleccionamos para incluir en dicha lista. Respecto a la reducción de la dimensionalidad, hemos aplicado *L-Average discrimination* para descartar aquellas expresiones cuyo valor está por debajo de la media de repeticiones de cada categoría.

En cada tabla observamos cuatro filas, que corresponden con diferentes consideraciones en el tipo y ponderación de las características utilizadas. En lo que resta del capítulo, emplearemos el término NEWPAR para hacer referencia a nuestra aproximación para la selección y cálculo de pesos:

- **SVM+BOW+TFIDF**. Hemos empleado la bolsa de palabras (BOW) ponderadas mediante TFIDF.
- **SVM+BOW+Pond. NEWPAR**. Las características se han valorado según el cálculo de pesos obtenido mediante NEWPAR.
- **SVM+BOW (Sust.)+Pond. NEWPAR**. La cantidad de expresiones seleccionadas de cada documento se ha reducido, de forma que ahora sólo utilizamos sustantivos.
- **SVM+N-GRAMAS+Pond. NEWPAR**. Empleamos sintagmas nominales como características, y además aplicamos la ponderación NEWPAR.

Presentamos los valores repartidos en 4 tablas, cada una mostrando las cifras obtenidas en diferentes corpus: Reuters (10 y 90 categorías) y OHSUMED (23 y 102 categorías). Como se puede observar en estos experimentos, los mejores resultados se obtienen al utilizar el esquema de selección y ponderación de características NEWPAR sobre n-gramas construidos con sintagmas nominales.

Comparativa con trabajos previos

Las tablas 5.8, 5.9 y 5.10 muestran los resultados obtenidos utilizando SVM en trabajos anteriores seguidos de los resultados conseguidos después de aplicar nuestra propuesta de selección y ponderación de características y emplear los mismos conjuntos de entrenamiento

	OneErr	IsErr	AvgP	BEP	maxF1	Cov
SVM+BOW+TFIDF	11,88	16,22	90,98	84,17	92,70	2,53
SVM+BOW+Pond. NEWPAR	**8,64**	**12,58**	93,16	87,25	94,45	2,43
SVM+BOW (Sust.)+Pond. NEWPAR	9,00	12,91	92,91	87,12	94,23	2,54
SVM+N-GRAMAS+Pond. NEWPAR	8,82	12,64	**93,46**	**87,52**	**94,83**	**2,15**

Tabla 5.5: Valoraciones micropromediadas al utilizar diferentes características y ponderaciones sobre el corpus Reuters, 90 categorías.

	OneErr	IsErr	AvgP	BEP	maxF1	Cov
SVM+BOW+TFIDF	30,26	53,53	72,59	62,67	77,19	4,15
SVM+BOW+Pond. NEWPAR	28,51	51,39	74,07	64,22	78,41	3,96
SVM+BOW (Sust.)+Pond. NEWPAR	27,42	50,20	74,95	65,00	79,17	3,92
SVM+N-GRAMAS+Pond. NEWPAR	**23,30**	**46,06**	**78,34**	**68,60**	**82,11**	**3,65**

Tabla 5.6: Valoraciones micropromediadas al utilizar diferentes características y ponderaciones sobre el corpus OHSUMED, 23 categorías (sub-árbol de enfermedades).

Universidad de Alicante

	OneErr	IsErr	AvgP	BEP	maxF1	Cov
SVM+BOW+TFIDF	28,35	41,76	74,61	65,08	78,94	**10,62**
SVM+BOW+Pond. NEWPAR	25,05	37,43	77,11	69,11	81,20	11,61
SVM+BOW (Sust.)+Pond. NEWPAR	28,13	41,66	74,33	65,08	78,68	12,74
SVM+N-GRAMAS+Pond. NEWPAR	**23,54**	**36,56**	**78,46**	**69,90**	**82,42**	12,11

Tabla 5.7: Valoraciones micropromediadas al utilizar diferentes características y ponderaciones sobre el corpus OHSUMED, 102 categorías (sub-árbol de enfermedades del corazón).

Algoritmo / Categoría	SVM lineal (BOW+MI) [Dumais et al., 1998]	SVM rbf (BOW+TFIDF) [Antonie, 2002]	kNN (bigramas+OR) [Tesar et al., 2006]	SVM lineal (NEWPAR)
earn	98,0	**98,4**	-	98,3
acq	93,6	95,2	-	**95,8**
money-fx	74,5	75,4	-	**83,2**
grain	94,6	91,8	-	**96,0**
crude	88,9	88,7	-	**91,5**
trade	75,9	77,3	-	**87,2**
interest	77,7	75,4	-	**79,4**
ship	85,6	86,6	-	**88,8**
wheat	**91,8**	85,7	-	88,7
corn	90,3	85,2	-	**94,6**
Macro pr.	87,1	86,0	-	**90,4**
Micro pr.	92,0	86,3	91,7	**94,1**

Tabla 5.8: Resultados utilizando BEP macro y micropromediado para Reuters, seleccionando las 10 categorías con mayor número de muestras.

y prueba que se usan en dichos trabajos. Hemos realizado comparaciones con trabajos previos que usan bigramas o bolsa de palabras, utilizando los esquemas de ponderación TFxIDF, información mutua (MI) y tasa impar (OR). Los mejores valores se han resaltado con negrita.

La tabla 5.8 muestra los resultados obtenidos usando Reuters-21578 (ModApte Split). De manera global nuestro método proporciona mejores resultados que el resto de propuestas. Se muestra una mejora del 2,3 % en la medida BEP micropromediada respecto al mismo algoritmo utilizando BOW. Además, el nuevo sistema basado en sintagmas nominales también obtiene mejores resultados que otros algoritmos de selección de características que consideran bigramas. Por ejemplo, se consigue una mejora de 2,6 % respecto a kNN utilizando bigramas como características y seleccionando las expresiones más relevantes mediante OR.

También en la evaluación local de los resultados se consigue una buena valoración. La tabla 5.8 muestra que se obtienen los mejores resultados en casi las 10 categorías con mayor número de muestras. Además, la medida BEP macropromediada muestra una mejora de 3,8 %. Esto indica que los resultados globales no se obtienen sólo por el incremento en la precisión y cobertura en unas categorías específicas.

Para comprobar exhaustivamente los resultados seguimos con la norma de utilizar diferentes corpus, y hemos realizado pruebas de evaluación también sobre el corpus OHSUMED. La tabla 5.9 muestra el valor BEP micropromediado calculado sobre las 23 categorías de enfermedades. NEWPAR obtiene una mejoría de 3,8 % frente a la propuesta de [Joachims, 1998], presentando también mayor precisión en casi todas las categorías individualmente.

También la evaluación mediante colecciones de documentos organizados en más categorías (90 para Reuters y 102 para OHSUMED) muestra incrementos en el rendimiento. La tabla 5.10

Algoritmo / Categoría	SVM rbf (BOW+TFIDF) [Joachims, 1998]	SVM lineal (NEWPAR)
Pathology	**58,1**	52,7
Cardiovascular	77,6	**80,9**
Immunologic	73,5	**77,1**
Neoplasms	70,7	**81,5**
Digestive system	73,8	**77,5**
Micro promedio (23 cat.)	66,1	**68,6**

Tabla 5.9: BEP para OHSUMED, sub-árbol de enfermedades, seleccionando las 5 categorías con mayor número de muestras, y calculando también el valor micropromediado sobre el conjunto total de las 23 categorías.

	Algoritmo/características / Corpus	SVM lineal (BOW+TFIDF) [Granitzer, 2003]	SVM lineal (NEWPAR)
Micro promedio	Reuters-90	86,4	**87,5**
	OHSUMED-102	63,2	**69,9**
Macro promedio	Reuters-90	45,2	**61,95**
	OHSUMED-102	50,3	**55,5**

Tabla 5.10: F1 para Reuters, 90 categorías, y para OHSUMED, 102 categorías (sub-árbol de enfermedades del corazón).

contiene la medida F1 calculada con micro y macro promedios utilizando las 90 categorías de Reuters y las 102 categorías del sub-árbol de enfermedades del corazón de OHSUMED. Para este último corpus se ha obtenido una mejora de más del 10 % (10,6 % para micropromedio y 10,3 % para macropromedio). En Reuters la mejoría es de 1,3 % y 37,1 % utilizando la medida F1 micro y macro promediada respectivamente, demostrando que la selección proporciona un mejor equilibrio entre clases.

Así pues, podemos concluir esta sección deduciendo que mediante el esquema propuesto de selección y ponderación de características se pueden conseguir incrementos apreciables en el rendimiento de clasificadores diseñados mediante SVM. Además, tales mejoras son sistemáticas, no siendo debidas a variaciones específicas en la cantidad de documentos o categorías, ni tampoco en la temática con la que se trabaje.

5.3.2. Ranking de categorías

En la sección anterior nos centramos en determinar la validez de los vectores de documentos generados, utilizando el esquema de ponderación NEWPAR para entrenar y evaluar el algoritmo SVM. A continuación procedemos a evaluar el proceso completo de clasificación, partiendo de las representaciones vectoriales y sumando los pesos de cada uno de los elementos de los vectores para generar el ranking de categorías, tal como describimos en la sección

4.4.2. Mostraremos diversas tablas que contienen los resultados obtenidos en trabajos previos junto con los resultados obtenidos aplicando el algoritmo de suma de pesos (ecuación 4.12). Para facilitar las comparaciones, en cada una de las tablas hemos resaltado en negrita los mejores resultados obtenidos. Además, para poder observar las variaciones de tamaño en la dimensionalidad del espacio vectorial, proporcionamos el número de características utilizado en cada caso.

Resultados con Reuters-21578

Las tablas 5.11, 5.12 y 5.13 muestran los resultados obtenidos utilizando el corpus Reuters-21578 (ModApte Split) como conjunto de pruebas.

En la tabla 5.11 observamos que en algunos casos, la reducción de dimensionalidad es notable para el corpus Reuters. Por ejemplo, en el conjunto de 90 categorías, si comparamos el resultado obtenido mediante el algoritmo SVM, vemos que podemos reducir el número de características un 33,3 % con respecto a éste, utilizando el esquema NEWPAR para la selección y ponderación de expresiones, y aún así, conseguimos una mejoría de casi 5 % medida con BEP. En [Joachims, 1998] se propone analizar el resultado de los algoritmos utilizando las mejores 500, 1.000, 5.000 o 10.000 características, valorándolas mediante ganancia de información (sección 3.3.2). El mejor resultado se observa en su caso utilizando 1.000 características para los algoritmos Rocchio, C4.5 y kNN, y todo el conjunto de características (9.962) para SVM y Bayes. La tabla 5.11 muestra las comparaciones con dichos valores.

De la tabla 5.12 podemos sacar conclusiones más específicas de la calidad del ranking de categorías devuelto por nuestra propuesta basada en la suma de pesos, al comparar con algoritmos tales como Rocchio, Perceptrón y MMP (*Multiclass Multilabel Perceptron*), estos dos últimos basados en redes neuronales (sección 2.3.4). Respecto a MMP, se trata de un algoritmo diseñado específicamente para conseguir un alto rendimiento en la obtención de rankings de categorías [Crammer y Singer, 2002]. Aunque MMP no requiere gran cantidad de recursos (el algoritmo recibe los documentos de entrenamiento de uno en uno), sí necesita actualizaciones continuas hasta completar todo el aprendizaje. La red neuronal debe refinarse mediante prueba y error, utilizando cada muestra de entrada y comparando el resultado obtenido con el esperado. Por ello, en [Crammer y Singer, 2003] se presentan por ejemplo las propuestas "Uniform" y "Rand", como posibles mejoras en el proceso de actualización de la red neuronal mediante cambios uniformes y aleatorios respectivamente. Además, también podemos observar las variantes "OneErr", "IsErr", "AvgP" y "maxF1", una para cada método de evaluación de la calidad del ranking de categorías durante el entrenamiento.

En la tabla 5.12 las comparaciones se han realizado con [Crammer y Singer, 2003], donde se han seleccionado 3.468 características de un total de más de 27.000. Para ello se ha empleado el algoritmo de Rocchio, calculando por cada categoría los pesos de cada característica única que se ha encontrado en todos los documentos de entrenamiento. El filtrado se ha realizado posteriormente, eligiendo un máximo de 100 palabras por categoría. Finalmente se obtiene un sólo conjunto de características mediante la unión de los conjuntos individua-

Algoritmo	Características	Variante	BEP
Bayes	9962	-	72,00
Rocchio	1000	β=1	79,90
C4.5	1000	-	79,40
kNN	1000	k=30	82,30
SVM	9962	polinomial combinado	86,00
	9962	rbf combinado	**86,40**
Suma de Pesos	6634	NEWPAR	**90,69**

Tabla 5.11: Comparación de resultados con los datos extraídos de [Joachims, 1998], utilizando BEP micro-promediada. El corpus utilizado es Reuters, 90 categorías.

les resultantes de procesar los documentos de cada categoría. Analizando los valores de las medida OneErr, deducimos que nuestra solución incrementa la validez de la decisión tomada en primera posición (la más probable), presentando un decremento de 14,5 a 8 en OneErr respecto a Rocchio. Además, las mejoras en los valores de IsErr, AvgP y maxF1 denotan también un incremento en la calidad de las categorías elegidas al descender en el ranking. Se presentan mejorías del orden del 5 % respecto a Rocchio y el mejor resultado obtenido mediante MMP medidas con AvgP y maxF1 respectivamente.

En la tabla 5.13 realizamos comparaciones con los resultados obtenidos mediante el algoritmo AdaBoost.MH [Cai y Hofmann, 2003], un tipo de clasificador unificado (ver sección 3.4.5). Empleamos el conjunto de 50 categorías con mayor número de muestras, y respecto al conjunto de características, se presentan dos variantes: "term" y "mixA". En la primera sólo se utilizan simples palabras como características, y en la segunda se añaden una serie de conceptos que permiten generar características semánticas. Con nuestra aproximación, el número de características se reduce en más de un 80 %, y la suma de pesos consigue una mejora de 4,6 % y 4,5 % en la precisión media (AvgP) y BEP respectivamente.

Observando los resultados generales, podemos deducir que utilizando la suma de pesos combinada con NEWPAR se obtiene una precisión por encima del 90 % para los subconjuntos del corpus Reuters-21578 empleados, mejorando los resultados obtenidos por los otros algoritmos presentados.

Resultados con OHSUMED

Las tablas 5.14, 5.15 y 5.16 muestran los resultados obtenidos al utilizar el corpus OHSUMED como colección de entrenamiento y prueba. A continuación procedemos a comentar los datos más destacables que se incluyen en cada una de ellas.

En la tabla 5.14 observamos los resultados al aplicar el esquema de selección y ponderación NEWPAR sobre los documentos del subconjunto de enfermedades [Joachims, 1998]. Si calculamos el ranking de categorías mediante la suma de pesos, obtenemos los mejores resultados: un 7,5 % de mejora respecto a SVM utilizando el esquema TFxIDF y midiendo la precisión

Algoritmo	Características	Variante	OneErr	IsErr	AvgP	maxF1
Rocchio	3468	-	14,48	18,82	89,73	84,53
Perceptrón	3468	-	9,59	15,71	90,50	88,56
MMP	3468	l.1	8,65	-	94,00	90,00
	3468	l.2	13,42	-	91,00	86,00
	3468	l.3	14,51	-	90,00	84,00
MMP (Uniform)	3468	OneErr	8,63	15,14	92,79	89,64
	3468	IsErr	8,65	13,07	93,63	90,32
	3468	AvgP	8,20	12,72	93,93	90,75
	3468	maxF1	8,23	**12,36**	**94,14**	90,80
MMP (Rand)	3468	OneErr	8,18	14,48	93,08	90,12
	3468	IsErr	8,18	12,52	93,89	90,81
	3468	AvgP	**8,11**	12,60	94,11	**90,84**
	3468	maxF1	8,56	13,09	93,75	90,38
Suma de Pesos	6634	NEWPAR	**8,09**	**11,85**	**94,29**	**95,56**

Tabla 5.12: Comparación de resultados con los datos extraídos de [Crammer y Singer, 2002, 2003]. El corpus utilizado es Reuters, 90 categorías.

Universitat d'Alacant
Universidad de Alicante

Algoritmo	Características	Variante	AvgP	BEP
AdaBoost.MH	37926	term	88,49	85,79
	42526	mixA	**90,49**	**86,98**
AdaBoost.MR	37926	term	88,14	84,33
	42526	mixA	90,23	86,07
Suma de Pesos	7136	NEWPAR	**94,68**	**90,90**

Tabla 5.13: Comparación de resultados con los datos extraídos de [Cai y Hofmann, 2003]. El corpus utilizado es Reuters, seleccionando las 50 categorías con mayor número de muestras.

Algoritmo	Características	Variante	BEP
Bayes	15561	-	57,00
Rocchio	15561	β=1,0	56,60
C4.5	500	-	50,00
kNN	15561	k=45	59,10
SVM	15561	poly. comb.	65,90
	15561	rbf. comb.	**66,00**
Suma de Pesos	17566	NEWPAR	**70,95**

Tabla 5.14: Comparación de resultados con los datos extraídos de [Joachims, 1998]. El corpus utilizado es OHSUMED, 23 categorías (sub-árbol de enfermedades).

con BEP. La cantidad de características utilizadas es similar (nosotros utilizamos aproximadamente un 15 % más). En [Joachims, 1998] se emplean simples palabras como características, se eliminan aquellas que aparecen menos de tres veces, y se eliminan las palabras de parada. Además se realiza un proceso de selección de términos utilizando ganancia de información, mediante el mismo procedimiento y criterios descritos en [Yang y Pedersen, 1997].

En la tabla 5.15 utilizamos el sub-árbol de enfermedades del corazón como colección de pruebas [Granitzer, 2003]. Se presenta una reducción de dimensionalidad de 62,1 % y además, comparando con el algoritmo de suma de pesos se produce una mejora en el rendimiento de 17,8 % medida con OneErr y 4,5 % empleando AvgP respecto al mejor de los resultados obtenido con SVM mediante la ponderación TFxIDF. La gran diferencia en la cantidad de características respecto a [Granitzer, 2003], se debe a que dicho autor no aplica métodos de selección de características específicos. Utiliza sin embargo técnicas de procesamiento habituales: elimina palabras de parada, reduce las palabras a su raíz mediante el algoritmo de Porter, y por último descarta aquellas que no aparecen en al menos 3 documentos.

En la tabla 5.16 evaluamos y comparamos el comportamiento obtenido al utilizar el subconjunto de las 50 categorías con mayor número de muestras. Empleamos como referencia el algoritmo AdaBoost.MH [Cai y Hofmann, 2003]. En este caso observamos un incremento de 37,1 % en la precisión medida con AvgP y un incremento de 5,7 % al medirla con BEP. Conviene destacar el notable incremento en el número de características, ya que se emplean casi un 50 % más. En [Cai y Hofmann, 2003] se eliminan una lista fija de palabras de parada, y además, también se descartan aquellas palabras que aparecen en menos de 10 documentos. Esta consideración conduce a un criterio de selección de características más restrictivo de lo habitual. El resultado son un total de 19.066 características, frente a las 28.337 propuestas en nuestro caso.

Como primera conclusión podemos observar que al igual que sucedió en Reuters, para todos los subconjuntos de prueba del corpus OHSUMED, el uso del esquema NEWPAR junto con la suma de pesos produce una mejora en la calidad de los resultados respecto a los obtenidos con el resto de algoritmos. Aunque en algunos casos incrementamos el número de características generado durante el entrenamiento, a primera vista, esto no significa un deterioro en la calidad del conjunto.

Algoritmo	Características	Variante	OneErr	AvgP
BoosTexter.RV	12428	β=avg	78,75	20,10
	12428	β=no	88,34	14,70
	12428	β=yes	29,21	77,40
Centroid Booster	12428	β=avg	55,23	56,20
	12428	β=no	88,47	15,30
	12428	β=yes	27,31	77,90
SVM light	12428	β=no	26,83	**79,00**
	12428	β=yes	**26,14**	73,90
Suma de Pesos	4706	NEWPAR	**21,49**	**82,58**

Tabla 5.15: Comparación de resultados con los datos extraídos de [Granitzer, 2003]. El corpus utilizado es OHSUMED, 102 categorías (sub-árbol de enfermedades del corazón).

Universitat d'Alacant
Universidad de Alicante

Algoritmo	Características	Variante	AvgP	BEP
AdaBoost.MH	19066	term	44,82	47,41
	20066	1000A	**47,03**	**49,70**
AdaBoost.MR	19066	term	43,98	47,41
	20066	1000A	45,97	49,52
Suma de Pesos	28337	NEWPAR	**64,52**	**52,53**

Tabla 5.16: Comparación de resultados con los datos extraídos de [Cai y Hofmann, 2003]. El corpus utilizado es OHSUMED, artículos publicados en el año 1987, 50 categorías con mayor número de muestras.

5.4. Relevancia de los parámetros en el proceso de clasificación

En esta sección vamos a analizar la contribución individual de cada uno de los parámetros considerados en el esquema NEWPAR, para observar la relevancia de los mismos, y confirmar de este modo los efectos positivos conseguidos al aplicar cada una de las intuiciones planteadas en esta tesis. Para ello obtendremos primero las medidas F1, AvgP, BEP y maxF1 micropromediadas utilizando el algoritmo original. Después omitiremos uno por uno cada parámetro y volveremos a evaluar de nuevo el proceso de clasificación. El objetivo es calcular finalmente el porcentaje de deterioro que produce la eliminación de cada parámetro sobre dichas medidas, viendo de esta manera su aportación.

Los gráficos de las figuras 5.3, 5.4, 5.5 y 5.6 indican la relevancia de cada parámetro para el proceso global de clasificación. Muestran el porcentaje de deterioro en el rendimiento para los corpus Reuters y OHSUMED cuando se elimina cada parámetro del algoritmo. Además, hemos utilizado una línea discontinua para representar los resultados promediados de los distintos corpus. Utilizaremos las siguientes etiquetas para describir las condiciones aplicadas para las diferentes evaluaciones:

- **Frases**: En la propuesta original partíamos de características formadas con n-gramas construidos mediante sintagmas nominales. Ahora pretendemos observar el comportamiento del algoritmo al utilizar como características palabras en vez de frases. Para ello utilizaremos n-gramas de longitud 1, de forma que los términos estarán compuestos por una sola palabra, que deberá ser sustantivo o adjetivo.
- **Diccionario, Tagger**: Si prescindimos del diccionario, no reduciremos las palabras a su raíz. Respecto al etiquetador (*tagger*), al descartar su funcionalidad sólo tendremos en cuenta aquellos sustantivos o adjetivos que vengan indicados en el diccionario, sin llevar a cabo ningún análisis del contexto u oración en el que aparece cada palabra.
- **Frases.PoS**: Las expresiones están formadas por una sola palabra de cualquier tipo. No utilizaremos por lo tanto ni diccionario ni etiquetador. Mediante este ajuste prescindimos de la habilidad de poder distinguir entre los distintos tipos de palabras, y todas se considerarán válidas (verbos, artículos, etc.) a no ser que se encuentren dentro de la lista de palabras de parada que se haya generado.
- **Títulos**: Las expresiones en los títulos tienen el mismo peso que el resto, mientras que en el algoritmo original, los títulos tenían el doble de valor.
- **Des.Cat.**: Los descriptores de las categorías no tienen influencia en la ponderación de las expresiones. No consideraremos más importantes aquellos términos que se encuentren incluidos en las expresiones que identifican las clases.
- **TF, DF, TF-DF, TFnew, CF, TL**: Para observar la influencia de cada uno de estos parámetros, procederemos a anularlos individualmente uno a uno para que los respecti-

Universitat d'Alacant
Universidad de Alicante

vos TFn_{ij}, DFn_{ij}, $TFnew_j$, CF_j y TLn_j no tengan ningún efecto durante el proceso de ponderación. Esto se consigue modificando la ecuación dada en la sección 4.2.5 para omitir en cada caso el valor indicado, calculando el peso de cada característica de forma diferente, tal como se indica a continuación:

$$TF \Rightarrow w_{ij} = \frac{(DFn_{ij} \cdot TLn_{ij} \cdot TFnew_j)}{CF_j}$$

$$DF \Rightarrow w_{ij} = \frac{(TFn_{ij} \cdot TLn_{ij} \cdot TFnew_j)}{CF_j}$$

$$TF - DF \Rightarrow w_{ij} = \frac{TLn_{ij} \cdot TFnew_j}{CF_j}$$

$$TFnew \Rightarrow w_{ij} = \frac{(TFn_{ij} + DFn_{ij}) \cdot TLn_{ij}}{CF_j}$$

$$CF \Rightarrow w_{ij} = (TFn_{ij} + DFn_{ij}) \cdot TLn_{ij} \cdot TFnew_j$$

$$TL \Rightarrow w_{ij} = \frac{(TFn_{ij} + DFn_{ij}) \cdot TFnew_j}{CF_j}$$

Recordemos que las frecuencias de término y documento (TFn_{ij}, DFn_{ij}) se refieren a los valores normalizados respecto a cada término t_j para la categoría c_i. La frecuencia de categoría (CF_j) identifica la cantidad de clases que contienen un término específico t_j dentro del conjunto de expresiones representativas. Respecto a TLn_j, nos servirá para tener en cuenta la cantidad de caracteres que contiene el término t_j.

Universitat d'Alacant
Universidad de Alicante

5.4.1. Ponderación y selección de características

Observando la figura 5.3 podemos valorar la influencia de los distintos parámetros respecto al proceso de construcción de los vectores de documentos. Para apreciar las variaciones en el rendimiento hemos escogido SVM como algoritmo de clasificación, y lo hemos entrenado utilizado el esquema NEWPAR como método de ponderación y selección de características, tal como hicimos en la sección 5.3.1.

De entre todas las medidas, eliminar TF-DF ocasiona el mayor deterioro en el rendimiento del clasificador, lo que significa que las frecuencias de términos y documentos tienen un significado crucial, tal como se ha publicado en numerosos trabajos anteriores.

Al incrementar la valoración de las expresiones que aparecen en los títulos también conseguimos mejoras importantes en el esquema de ponderación, pues como se observa en la gráfica, es el segundo parámetro más relevante.

El uso de frases o sintagmas nominales en vez de palabras individuales de cualquier tipo también mejora los resultados significativamente para OHSUMED. Si observamos la tabla 5.17 vemos que los incrementos graduales en el número de características vienen acompañados

de una mejor valoración de la clasificación final. Por ejemplo, observamos que F1 se incrementa de 63,9 a 68,6 al pasar de 2.055 a 24.823 características. Sin embargo no se produce el mismo efecto en Reuters. Tal como confirman los resultados expuestos en la tabla 5.17, los documentos del corpus Reuters se representan mejor mediante simples palabras sin stemming. Aunque se pueda pensar que un incremento en el número de características siempre debe venir acompañado por un incremento en la precisión, en el caso de esta última colección de documentos, esto no es así. Por ejemplo, vemos que mediante la simple bolsa de palabras obtenemos un valor muy alto de F1 utilizando sólo 896 características. Al emplear sintagmas nominales, incrementamos dicho conjunto a un total de 4.745 características, pero ocasionamos contradictoriamente una leve disminución en la calidad del proceso de clasificación. Por ello, aunque los resultados aquí expuestos superan a los presentados en la bibliografía, no podemos deducir que tales incrementos son debidos únicamente al uso de sintagmas nominales, pues en el caso de Reuters, el incremento en la cantidad de características o de la información representada por las mismas, no se ve reflejado en una mejoría del proceso de clasificación.

La frecuencia de categorías (CF) es el cuarto parámetro en orden de relevancia. Esto corrobora la intuición de que la capacidad discriminativa de una expresión es mayor cuanto menor sea el número de categorías a las que representa.

Respecto a la frecuencia de aparición de los términos en los nuevos documentos ($TFnew$), observamos un efecto positivo para ambos corpus, teniendo una contribución de aproximadamente un 1,5 %. Debemos tener en cuenta que se trata de un parámetro cuya obtención no reviste complejidad alguna, y se puede calcular directamente en la recepción de los nuevos textos que se reciban para ser clasificados.

En último lugar observamos que factores tales como el incremento en la ponderación para aquellas expresiones que coincidan con los descriptores de categorías, o el uso de las longitudes de los términos, o de un diccionario o etiquetador, son los factores menos relevantes para SVM.

Todo esto nos lleva a la conclusión de que para homogeneizar los resultados independientemente del corpus con el que se trabaje y el algoritmo de clasificación que se vaya a emplear, no siempre resulta beneficioso incrementar el número de características o la cantidad de información que puedan contener, sino que además debemos prestar especial atención también a la forma de calcular las ponderaciones de las características, tanto para realizar un filtrado previo, como para su uso final en las representaciones vectoriales de los documentos.

5.4.2. Ranking de categorías

Los gráficos 5.4, 5.5 y 5.6 indican la relevancia de cada parámetro para el proceso de clasificación mediante ranking de categorías, utilizando el algoritmo de suma de pesos. Muestran el porcentaje de deterioro en el rendimiento medido con AvgP, BEP y maxF1 cuando se elimina cada parámetro del algoritmo. Se puede consultar el apéndice C para obtener más detalles de los resultados exhaustivos obtenidos empleando éstas y otras medidas para la evaluación.

Al igual que en la gráfica anterior (Figura 5.3), en la que utilizábamos SVM, el uso de las

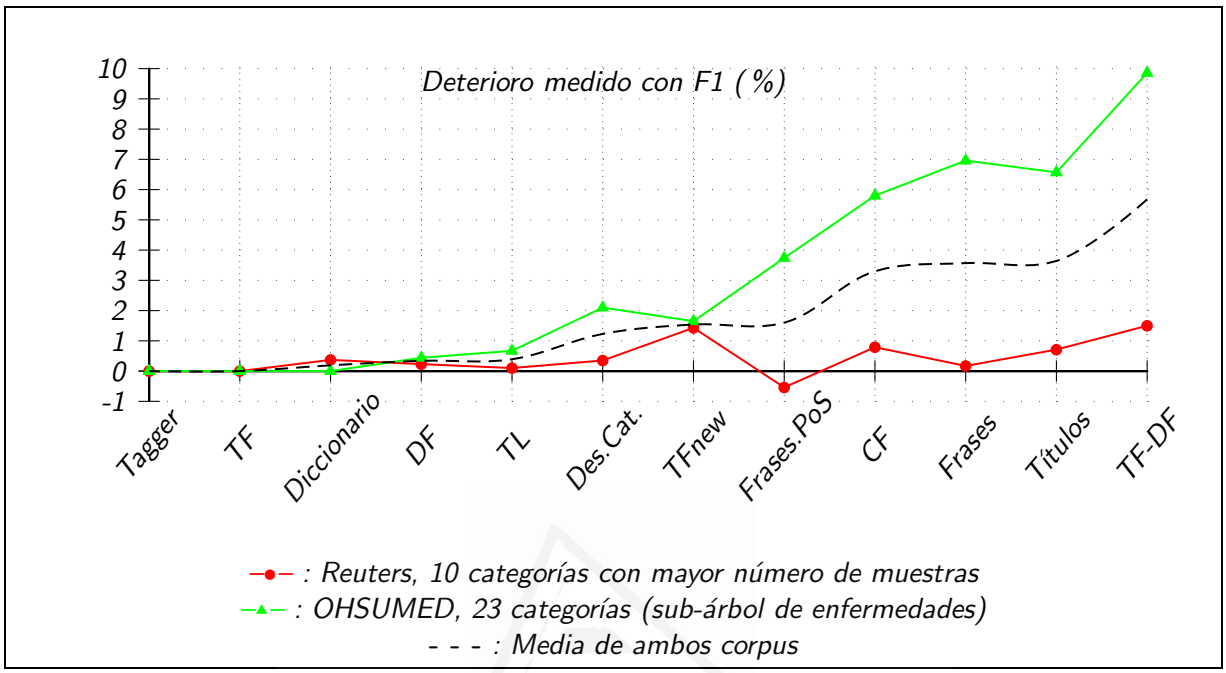


Figura 5.3: Deterioro en el rendimiento de clasificación binaria después de quitar cada parámetro. Los parámetros se han colocado de izquierda a derecha en orden creciente, del que menos afecta al que más (según la media de ambos corpus).

Universitat d'Alacant
Universidad de Alicante

		Número de características	F1
Sustantivos	Reuters-10	721	94,0
	OHSUMED-23	2055	63,9
Cualquier palabra	Reuters-10	896	**94,7**
	OHSUMED-23	2221	66,0
Sintagmas nominales	Reuters-10	4745	94,2
	OHSUMED-23	24823	**68,6**

Tabla 5.17: Número de características en relación con F1 micropromediada. Los resultados se han obtenido para Reuters, seleccionando las 10 categorías con más muestras, y para OHSUMED, 23 categorías (sub-árbol de enfermedades).

frecuencias de términos y documentos (TF-DF) sigue siendo determinante en el rendimiento del clasificador.

Si descartamos el uso de frases también observamos un gran deterioro en los resultados calculados sobre la media de los corpus, teniendo mayor efecto sobre OHSUMED. De este comportamiento deducimos que si incrementamos por norma las longitudes de las expresiones, podemos conseguir que éstas almacenen más información, y puedan influir directamente en el ranking de categorías.

También queda reflejado que es importante considerar la frecuencia de categorías como factor discriminativo, y que además resulta beneficioso utilizar simplemente sustantivos y adjetivos en la construcción de expresiones. Al aplicar estas consideraciones no sólo conseguimos almacenar los conceptos sin pérdida de información, sino que además filtraremos expresiones que pueden causar interferencias en las decisiones de clasificación.

Hay que tener en cuenta que el tipo de características propuestas en esta tesis, y los métodos de selección y ponderación descritos, se han elegido especialmente para que se generen vectores de documentos lo más autocontenidos posibles. El objetivo no ha sido sólo valorar si una expresión es relevante o no, o si una expresión es más relevante que otra. Pretendemos generar características que reflejen la relación que pueda existir con las categorías a las que pertenecen los documentos que las contienen. Por ello observamos que al sumar la valoración de los parámetros propuestos, se consiguen ordenar dichas categorías para un nuevo texto con mayor eficiencia que otros algoritmos de clasificación. En el caso de la suma de pesos, factores tales como la frecuencia de documento o de término pierden valor de forma individual (aparecen en el último lugar de la gráfica), pues no contienen información específica que nos ayude a discriminar entre las clases del corpus.

Universitat d'Alacant
Universidad de Alicante

5.4.3. Peso óptimo para las expresiones de los títulos

Tal como sugeríamos en la sección 4.2.1, puede resultar conveniente considerar las expresiones en los títulos más importantes que el resto. Nuestra propuesta inicial para llevar a la práctica tal idea se basaba en doblar las frecuencias TF y DF (ver sección 4.2.4), pero no proporcionábamos ninguna explicación de porqué valorar tales expresiones utilizando dicho factor y no cualquier otro. A continuación justificaremos el porqué de esa decisión llevando a cabo diversas pruebas que nos permitan observar el comportamiento del clasificador aplicando diferentes factores de incremento antes de calcular y sumar los pesos que nos proporcionen el ranking de categorías.

El gráfico 5.7 muestra las variaciones en la precisión proporcionada por el algoritmo de suma de pesos después de aplicar cinco factores diferentes para las expresiones en los títulos, esto es, multiplicando las frecuencias por 1, 2, 3, 4 y 5, valores que vendrán representados por las etiquetas "x1", "x3", "x4", y "x5" respectivamente (el valor "x2" no se representa, pues es el considerado como base para establecer la comparación). Los valores positivos mostrados indican decrementos en la calidad de los resultados al utilizar valores distintos de 2. Aunque se

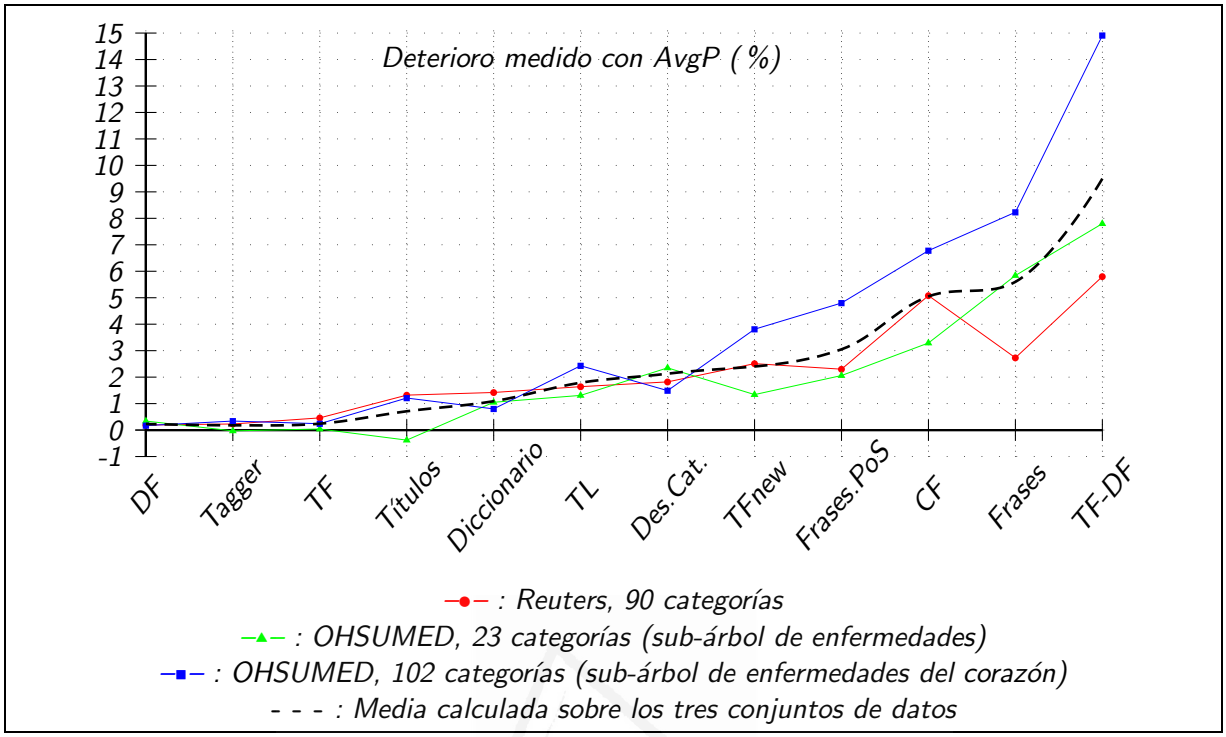


Figura 5.4: Deterioro en el rendimiento del ranking de categorías medido con AvgP después de quitar cada uno de los parámetros. Los parámetros aparecen de izquierda a derecha en orden creciente, del menor al más relevante (según la media de los tres corpus).

obtienen diferentes resultados al utilizar diferentes conjuntos de datos, las medias indican que el mejor rendimiento se obtiene cuando se duplican las frecuencias TF y DF. En el apéndice D se pueden obtener más detalles sobre los resultados obtenidos mediante diferentes medidas de evaluación, y los correspondientes porcentajes de deterioro al aplicar cada uno de los factores.

Universidad de Alicante

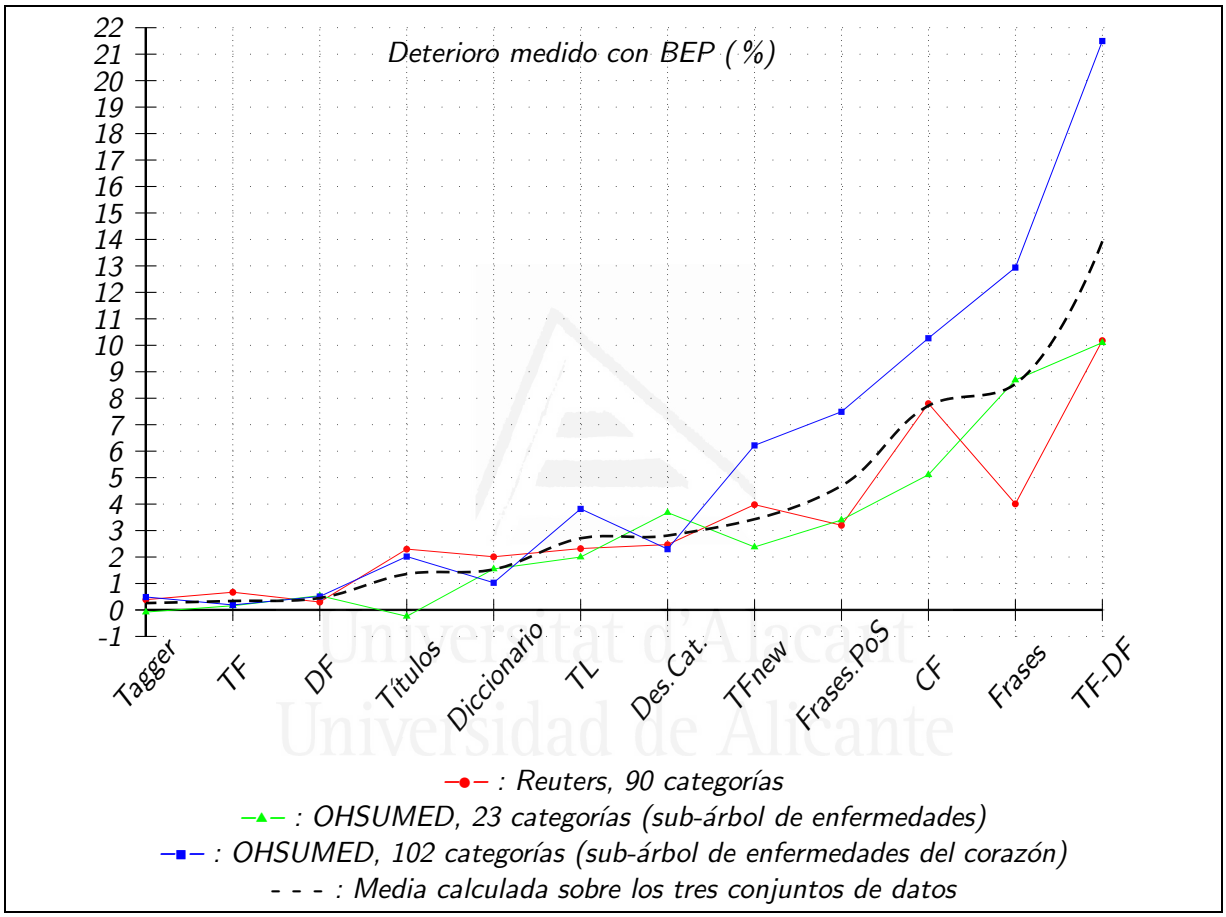


Figura 5.5: Deterioro en el rendimiento del ranking de categorías medido con BEP después de quitar cada uno de los parámetros. Los parámetros aparecen de izquierda a derecha en orden creciente, del menor al más relevante (según la media de los tres corpus).

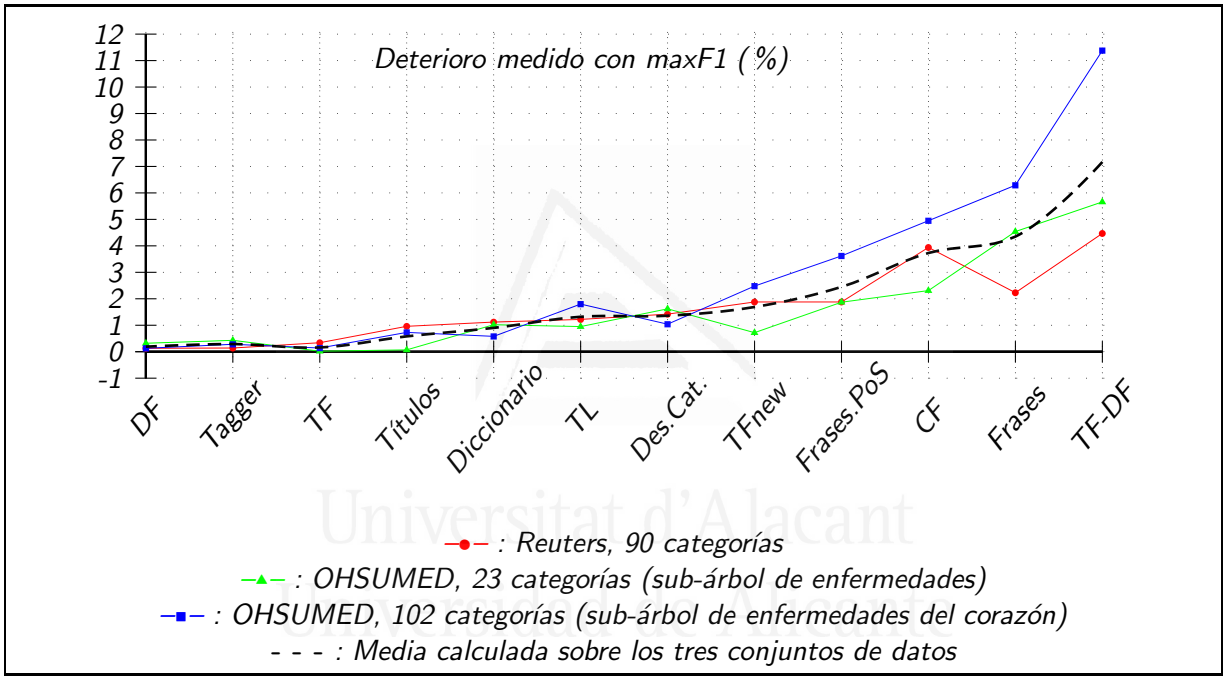


Figura 5.6: Deterioro en el rendimiento del ranking de categorías medido con maxF1 después de quitar cada uno de los parámetros. Los parámetros aparecen de izquierda a derecha en orden creciente, del menor al más relevante (según la media de los tres corpus).

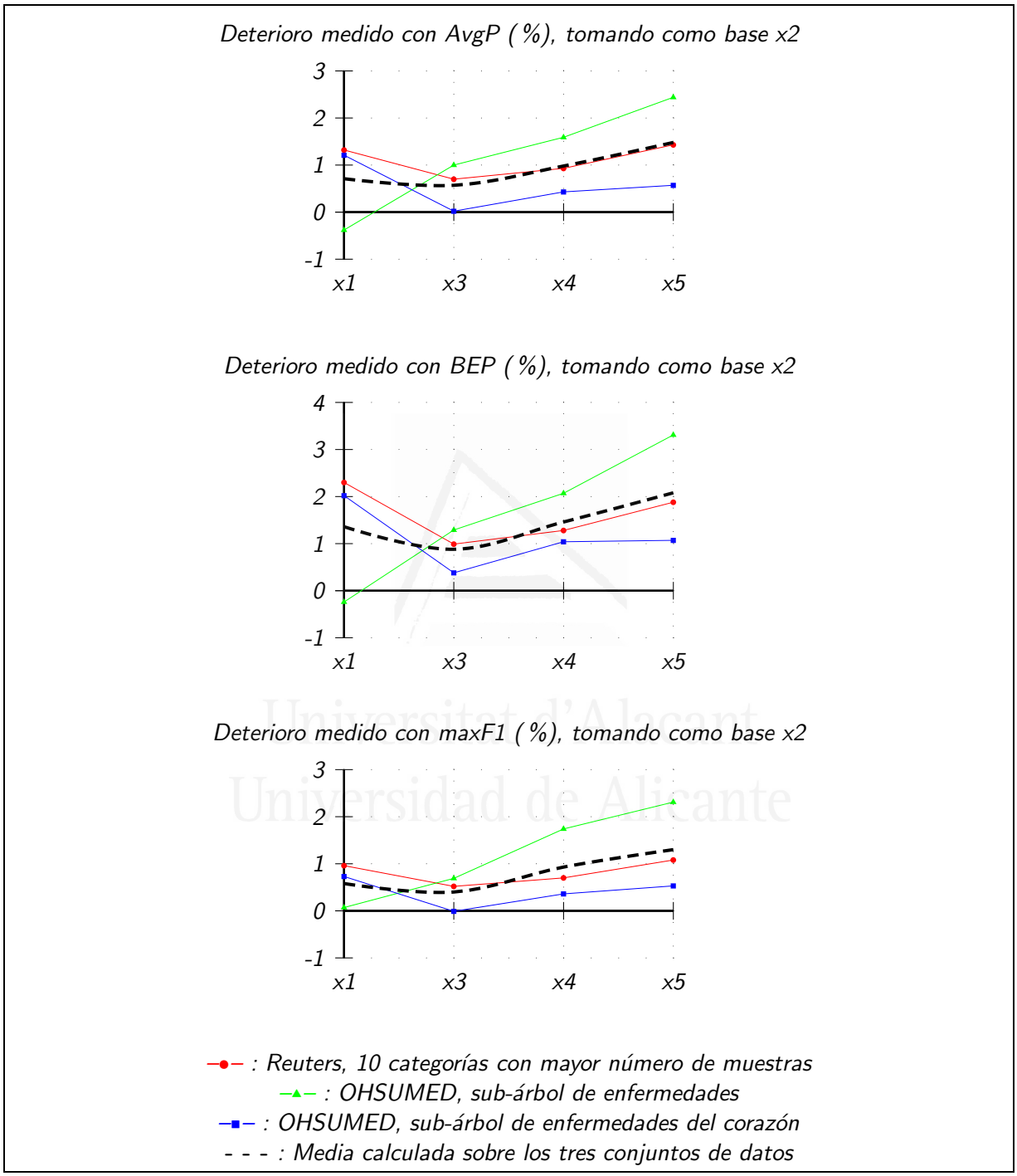


Figura 5.7: Deterioro en el rendimiento aplicando diferentes factores de ponderación para los títulos de los documentos.

Universitat d'Alacant
Universidad de Alicante

Universitat d'Alacant
Universidad de Alicante

Capítulo 6

Aplicaciones

En este capítulo vamos a describir diferentes aplicaciones de nuestro sistema de clasificación de textos. En primer lugar veremos cómo se puede utilizar toda la información textual disponible en el historial médico de un paciente para intentar ofrecer una lista ordenada de categorías que puedan aportar una orientación sobre las posibles enfermedades asociadas con los síntomas descritos. Mostraremos casos reales, asociados con los diagnósticos proporcionados por expertos, realizando comparaciones respecto a las respuestas obtenidas por nuestro sistema.

En la última parte del capítulo además realizaremos otras propuestas de aplicación también dentro del ámbito médico, que pueden ser abordadas mediante la clasificación de textos utilizando el algoritmo descrito. Nos podemos encontrar, por ejemplo, con la posibilidad de obtener listados de principios activos que nos conduzcan a posibles medicamentos que puedan ser prescritos dependiendo de unos síntomas proporcionados como entrada. Veremos también cómo podemos enfocar dicha aplicación a la obtención de plantas medicinales.

Finalmente, expondremos cómo utilizar el diseño jerárquico de los descriptores MeSH para orientar a un usuario mediante términos relacionados que le permitan refinar una búsqueda. También, de la misma forma, veremos cómo construir un directorio médico con el propósito de localizar y agrupar aquellos artículos de la Wikipedia que versen sobre temas médicos.

6.1. Diagnóstico médico

A continuación describimos una aplicación real para asistir diagnósticos médicos, que se basa en el uso exclusivo de técnicas de aprendizaje automático para obtener ranking de categorías. Mediante el procesamiento de una gran cantidad de artículos biomédicos con el propósito de entrenar el algoritmo de categorización, pretendemos dotarlo con la habilidad de asociar síntomas a descriptores MeSH. Esto significa que si disponemos de un clasificador con la capacidad suficiente para procesar una rama completa del árbol, podemos dotarle de

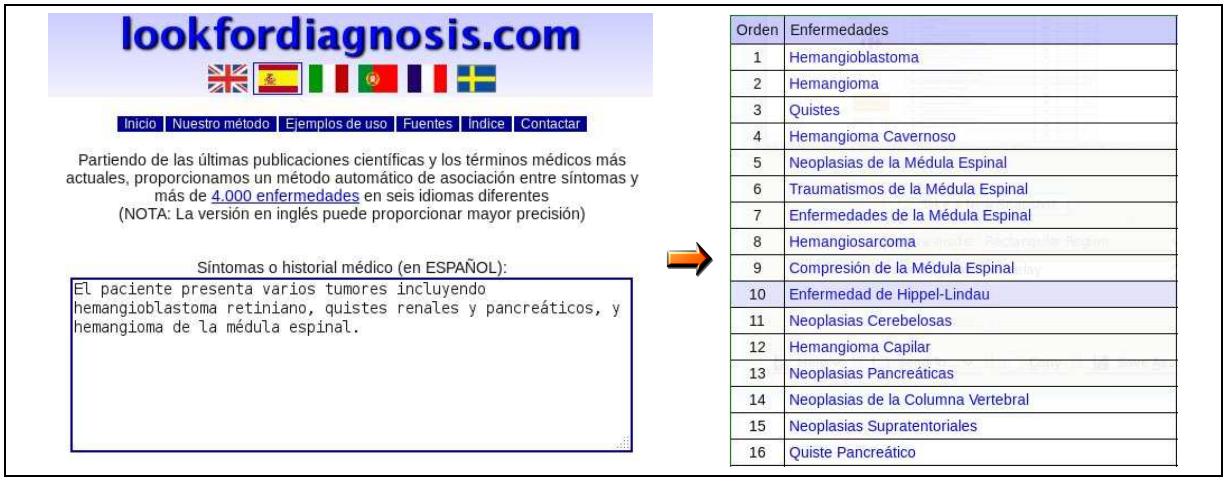


Figura 6.1: Ejemplo de diagnóstico obtenido mediante ranking de categorías. A la izquierda se introducen diversos síntomas del síndrome de Von Hippel-Lindau (hemangioblastoma retiniano, quistes renales y pancreáticos, hemangioma de la médula espinal, etc.), y a la derecha se muestra el resultado proporcionado por el clasificador.

la habilidad de asociar síntomas con enfermedades o tratamientos. En primera instancia y gracias a la disponibilidad de documentos que nos permiten realizar evaluaciones objetivas, proponemos realizar pruebas dentro del ámbito del diagnóstico médico, intentando asociar síntomas con enfermedades. Para intentar verificar la validez de esta propuesta, realizaremos experimentos con un conjunto de documentos de evaluación que contiene varios cientos de historiales clínicos reales.

Para interactuar con el clasificador, hemos desarrollado un interfaz web multilingüe (accesible a través de la dirección http://www.lookfordiagnosis.com) para que cualquier usuario pueda obtener fácilmente un ranking de diagnósticos partiendo de la información introducida en un cuadro de texto. Podemos observar un ejemplo de funcionamiento en la figura 6.1, donde se introducen los síntomas de un paciente y se obtiene un listado de enfermedades relacionadas.

En la actualidad, se están realizando muchos esfuerzos para conseguir automatizar el proceso de catalogación de artículos médicos mediante técnicas de clasificación de textos. Se han diseñado técnicas específicas para intentar identificar la temática de cada documento para incluirlos en la base de datos MEDLINE [Ibushi y Tsujii, 1999] sin intervención humana. Sin embargo, hasta la fecha, dicho proceso de clasificación no se ha utilizado para asistir en el diagnóstico médico partiendo de la información textual proporcionada por la literatura biomédica y los historiales médicos de los pacientes.

Cada año se añaden miles de documentos a las bases de datos *National Library of Medicine* y *National Institutes of Health* (PubMed[1]). La mayoría de ellos han sido indexados manualmente asignando a cada documento una o varias entradas de un vocabulario preestablecido conocido como MeSH (Medical Subject Headings). El árbol MeSH es una estructura

[1] http://www.pubmed.gov

jerárquica de términos médicos que se usan para definir el tema principal del que trata un artículo médico. Debido al amplio uso de esta terminología, se pueden encontrar traducciones a diversos idiomas. Por ejemplo, disponemos del vocabulario estructurado y trilingüe DeCS[2] (Descriptores en Ciencias de la Salud), que fue creado para indexar artículos de revistas científicas, libros, informes técnicos y otros tipos de materiales, así como para ser usado en la búsqueda y recuperación de literatura científica escrita en español, portugués e inglés. Para el diagnóstico médico se utilizará el conjunto de descriptores MeSH que identifican las enfermedades (de C01 a C23), que define más de 4.000 estados patológicos y ofrece la posibilidad de buscar casos documentados de cada uno de ellos.

Mediante la propuesta que aquí se plantea, se busca obtener una estimación de una lista ordenada de diagnósticos partiendo del historial médico de un paciente. Para abordar este problema, se propone utilizar el algoritmo de clasificación descrito previamente en el capítulo 4. De esta forma, una detallada descripción de los síntomas es suficiente para obtener una lista de posibles enfermedades, junto con una estimación de probabilidades (grado de confianza) y la bibliografía en la que se basan los resultados.

No hemos utilizado decisiones binarias de clasificación (relevante o no relevante), puesto que se podrían descartar algunas entradas interesantes del MeSH que posiblemente deberían ser tenidas en cuenta. Al optar por algoritmos de ranking de categorías, podremos obtener un listado ordenado de todos los posibles diagnósticos de forma que, finalmente, un experto pueda decidir cuál de ellos se ajusta mejor a la historia clínica.

El algoritmo de clasificación utilizado ha sido la suma de pesos (SOW) descrito en el capítulo 4, que resulta más adecuado que el resto por su eficiencia, precisión y capacidad de aprendizaje incremental. Puesto que las bases de datos médicas se actualizan con frecuencia y crecen continuamente, es preferible usar un método de aprendizaje rápido y desatendido que permita realizar actualizaciones fácilmente, sin ocasionar una degradación en el funcionamiento después de incrementar el número de categorías o la cantidad de muestras de aprendizaje. La restrictiva complejidad de otros clasificadores tales como SVM podrían derivar en un problema intratable [Ruch, 2005].

A continuación explicaremos la forma en la que se han desarrollado los experimentos, incluyendo una descripción completa de los métodos y las fuentes que hemos usado para obtener los datos de entrenamiento y evaluación. Después proporcionaremos varios ejemplos de historiales médicos de diversos pacientes y mostraremos los diagnósticos realizados por varios expertos, comparándolos con los proporcionados por el clasificador. Por último, valoraremos el rendimiento del clasificador en dicha tarea empleando diferentes medidas de evaluación.

6.1.1. Corpus de trabajo

Los datos de aprendizaje se han extraído de la base de datos PubMed, pues se trata de un motor de búsqueda de libre uso, que proporciona acceso a millones de citas y resúmenes de

[2] http://decs.bvs.br/I/homepagei.htm

artículos de investigacin biomédica. Hemos seleccionado cada caso registrado sobre cualquier enfermedad que haya sido escrito en inglés, que contenga resumen y título y que, además, esté relacionado con los seres humanos.[3] Estos documentos han sido obtenidos mediante la consulta "*diseases category[MAJR] OR mental disorders[MAJR]*", donde *[MAJR]* es la abreviatura de "*MeSH Major Topic*" y especifica la obtención sólo de aquellos documentos que versan principalmente sobre enfermedades. La consulta se puede ejecutar directamente desde la interfaz web accesible desde la página principal de PubMed, donde se puede observar la cantidad de artículos disponibles que cumplen las restricciones propuestas (510.240 documentos[4]). También desde la web se puede empaquetar y descargar el resultado de la consulta en un fichero. Esto nos proporciona más de medio millón de documentos asociados a 4.315 categorías MeSH, cada una de ellas con al menos una muestra para completar el aprendizaje.

Respecto al conjunto de prueba, hemos utilizado 400 historias médicas de la Escuela de Medicina de la Universidad de Pittsburgh (Departamento de Patología[5]). Aunque la web contiene más de 500 historias,[6] no todas ellas son válidas para la evaluación propuesta. Hay algunas que no proporcionan un diagnóstico concreto, sino sólo una discusión acerca del caso, y algunos otros no tienen una asociación directa al árbol MeSH. De cada documento hemos extraído el título y la historia clínica, incluyendo los hallazgos radiológicos y las descripciones macro y microscópicas. Toda esta información se encuentra ya estructurada en secciones y por lo tanto, su obtención es relativamente sencilla. Para conocer la salida esperada, se ha extraído el nivel más alto del árbol de categorías MeSH correspondiente a los diagnósticos dados (en el apéndice A, sección A.3, se pueden observar las agrupaciones de las categorías de la rama de enfermedades). Por ejemplo, un paciente diagnosticado con baja visión ("low vision"), tendrá una enfermedad del grupo de las oftalmopatías ("eye diseases"). Como se observa en la página web que contiene los casos, el diagnóstico final se proporciona en los archivos identificados con el nombre "dx.html", a los que se puede acceder mediante el hipervínculo etiquetado con "final diagnosis". Del contenido de dichos archivos sólo se ha tenido en cuenta el título, donde aparece claramente la enfermedad asignada por expertos médicos.

Universitat d'Alacant
Universidad de Alicante

6.1.2. Medidas de evaluación

Para evaluar la validez de este método, se han utilizado tres medidas comunes (ver sección 5.2): Precisión con cobertura 0 ($P_{r=0}$), precisión media ($AvgP$) y Break Even Point (BEP). A continuación se exponen los motivos que justifican la elección de dichas medidas:

- A veces, sólo un diagnóstico es válido para un paciente en particular. En estos casos,

[3] Debemos aclarar que podemos encontrar una gran cantidad de artículos médicos que tratan sobre enfermedades animales. En nuestro caso es importante filtrar tales documentos para centrar los diagnósticos en pacientes humanos.

[4] Cifra obtenida en febrero de 2007. En mayo de 2013 se obtienen 662.102 documentos al ejecutar la misma consulta.

[5] http://path.upmc.edu/cases.

[6] Cifra obtenida en febrero de 2007. En la actualidad (mayo de 2013) se dispone de más de 700 casos.

$P_{r=0}$ nos permite cuantificar las respuestas erróneas, puesto que indica la proporción de respuestas correctas colocadas en la posición más alta.

- Para valorar la calidad de la lista entera se utiliza AvgP, puesto que esta medida desciende por toda la lista promediando la precisión hasta que se cubren todas las posibles respuestas.
- BEP es el valor donde la precisión se equipara con la cobertura, esto es, cuando se considera el máximo número de enfermedades relevantes como el umbral de corte.

Por último, para poder realizar comparaciones con otros procesos de clasificación genéricos, hemos decidido seguir el mismo procedimiento de evaluación que el expuesto en [Joachims, 1998] y hemos agrupado las enfermedades para obtener su equivalente en el nivel más alto del árbol de enfermedades.

6.1.3. Funcionamiento rápido y funcionamiento por jerarquías

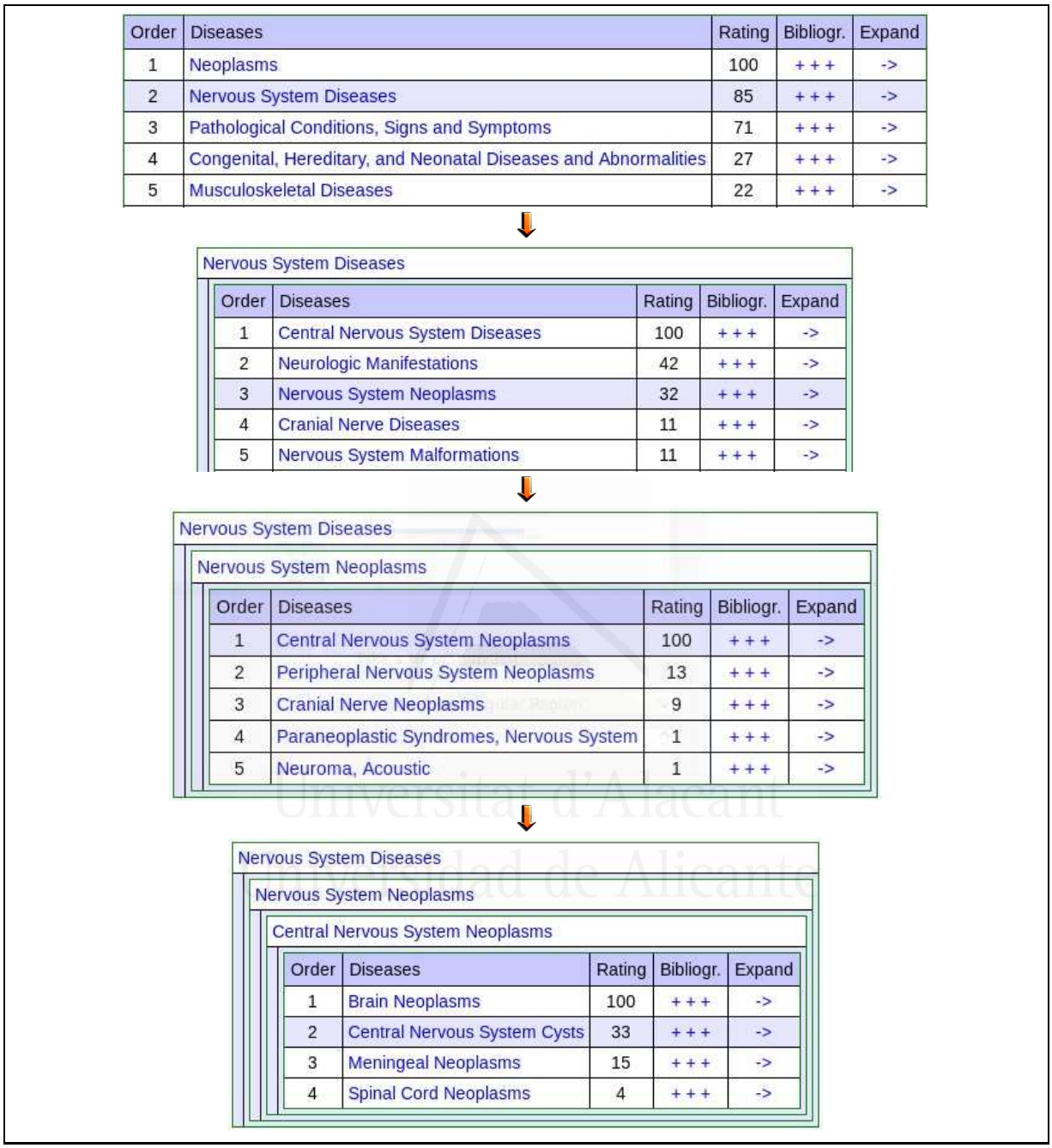
El árbol MeSH se encuentra estructurado en una jerarquía donde los términos más genéricos se encuentran cerca de la raíz, y los más específicos aparecen a medida que se desciende hacia niveles inferiores. Todos los términos, menos los que aparecen en el nivel más alto, pueden funcionar como nodos hoja, es decir, pueden ser asignados a un documento. Cada término aparece por lo menos en algún lugar de la estructura jerárquica, y además, es muy frecuente que aparezca en diferentes posiciones de la misma.

Para poder desplazarnos por las distintas ramas del árbol MeSH hemos desarrollado un modo de funcionamiento jerárquico, donde podemos descender desde un primer nivel de clasificación mientras se disponga de términos más específicos para la expansión, tal y como se muestra en la figura 6.2. El número de niveles no es fijo, pues diferentes caminos a través de la jerarquía tienen diferentes longitudes, pudiéndonos encontrar con un máximo de 11 decisiones para alcanzar algún nodo hoja. De esta forma, la interfaz permite la navegación entre niveles para que el usuario pueda ir bajando a través del árbol de términos MeSH, seleccionando las distintas ramas dependiendo de las probabilidades que se muestren y, principalmente, teniendo en cuenta sus propios criterios.

Además, se proporciona la posibilidad de obtener un diagnóstico rápido para obtener directamente una lista ordenada de todas las enfermedades posibles, tal como se muestra en la figura 6.3.

6.1.4. Entorno multilingüe

Con respecto al idioma, un corpus puede contener muestras de un único idioma (corpus monolingüe) o de varios idiomas (corpus multilingüe). En el caso de colecciones multilingües, la cantidad de información a almacenar se multiplica. Por ejemplo, nos podemos encontrar



Order	Diseases	Rating	Bibliogr.	Expand
1	Neoplasms	100	+ + +	->
2	Nervous System Diseases	85	+ + +	->
3	Pathological Conditions, Signs and Symptoms	71	+ + +	->
4	Congenital, Hereditary, and Neonatal Diseases and Abnormalities	27	+ + +	->
5	Musculoskeletal Diseases	22	+ + +	->

Nervous System Diseases

Order	Diseases	Rating	Bibliogr.	Expand
1	Central Nervous System Diseases	100	+ + +	->
2	Neurologic Manifestations	42	+ + +	->
3	Nervous System Neoplasms	32	+ + +	->
4	Cranial Nerve Diseases	11	+ + +	->
5	Nervous System Malformations	11	+ + +	->

Nervous System Diseases

Nervous System Neoplasms

Order	Diseases	Rating	Bibliogr.	Expand
1	Central Nervous System Neoplasms	100	+ + +	->
2	Peripheral Nervous System Neoplasms	13	+ + +	->
3	Cranial Nerve Neoplasms	9	+ + +	->
4	Paraneoplastic Syndromes, Nervous System	1	+ + +	->
5	Neuroma, Acoustic	1	+ + +	->

Nervous System Diseases

Nervous System Neoplasms

Central Nervous System Neoplasms

Order	Diseases	Rating	Bibliogr.	Expand
1	Brain Neoplasms	100	+ + +	->
2	Central Nervous System Cysts	33	+ + +	->
3	Meningeal Neoplasms	15	+ + +	->
4	Spinal Cord Neoplasms	4	+ + +	->

Universitat d'Alacant
Universidad de Alicante

Figura 6.2: Ejemplo de salida después de la expansión manual de categorías desde el primer nivel, correspondiente al historial médico de la figura 6.4. Si optamos por realizar un diagnóstico utilizando la jerarquía del MeSH, nuestro sistema proporciona a cada paso sólo descriptores de cada nivel, hasta llegar a los nodos hoja. En este ejemplo, observamos cómo el usuario ha ido refinando sus elecciones, desde las enfermedades del sistema nervioso (Nervous System Diseases), pasando por neoplasias del sistema nervioso (Nervous System Neoplasms), hasta llegar a las neoplasias del sistema nervioso central (Central Nervous System Neoplasms), donde se obtiene el diagnóstico final, que aparece en la segunda posición de la última imagen (Quistes del Sistema Nervioso Central - Central Nervous System Cysts).

Order	Diseases	Rating	Bibliogr.	Expand
1	Cysts	100	+ + +	->
2	Craniopharyngioma	72	+ + +	
3	Pituitary Neoplasms	49	+ + +	->
4	Hemosiderosis	41	+ + +	
5	Granuloma	39	+ + +	->
6	Headache	37	+ + +	
7	Brain Diseases	37	+ + +	->
8	Xanthomatosis	33	+ + +	->
9	Brain Neoplasms	29	+ + +	->
10	Soft Tissue Neoplasms	25	+ + +	->
11	Central Nervous System Cysts	24	+ + +	->
12	Hydrocephalus	23	+ + +	->
13	Fibroma	22	+ + +	->
14	Leukemia, Mixed-Cell	20	+ + +	
15	Pituitary Diseases	19	+ + +	->

Figura 6.3: Ejemplo de diagnóstico mediante un solo ranking donde todas las categorías aparecen en el mismo nivel. El diagnóstico corresponde al historial médico de la figura 6.4. La enfermedad diagnosticada por los expertos aparece en undécimo lugar (Quistes del Sistema Nervioso Central - Central Nervous System Cysts).

con corpus paralelos donde, para cada documento, se debe guardar su traducción literal, así como todos los detalles referentes a las alineaciones a nivel de documento, párrafo o sentencia entre los diferentes idiomas. También podemos encontrarnos con corpus menos estrictos, conocidos como comparables, que resultan más sencillos en su estructura pero menos útiles para aplicaciones reales, puesto que el número de documentos y términos suelen ser simplemente similares y la temática, aunque homogénea, puede quedar lejos de ser la misma para todos los idiomas.

Nuestro sistema está adaptado para su aplicación en entornos multilingües. Tal como hemos expuesto, todos los documentos utilizados durante el proceso de aprendizaje están escritos sólo en inglés. Por lo tanto, estamos utilizando un corpus monolingüe para el entrenamiento. Para conseguir que un nuevo documento que se proporcione en otro idioma diferente del inglés pueda ser aceptado, traduciremos el texto antes de ser utilizado como entrada del algoritmo de clasificación (la traducción se realizará de manera completamente automática y transparente al usuario). Aunque podamos perder cierta precisión en los resultados, este procedimiento nos evitará la necesidad de corpus paralelos, además de permitirnos utilizar el mismo clasificador para cualquier idioma mientras dispongamos del correspondiente traductor.

Puesto que todos los datos de aprendizaje están escritos en inglés, cada síntoma que se proporcione como entrada al algoritmo también debe estar escrito en inglés. Para conseguir que el usuario pueda introducir texto en cualquier idioma y el algoritmo lo reciba en inglés, utilizaremos el traductor estadístico MOSES [Koehn et al., 2007]. En primer lugar entrenaremos el traductor empleando el corpus EMEA [Tiedemann, 2009], una colección de casi 2.000 documentos proporcionados por la Agencia Europea de Medicamentos, disponibles en 22 idio-

mas y alineados de forma que se puede observar la correspondencia directa entre más de un millón de frases para cada uno de los idiomas. El vocabulario empleado resulta altamente beneficioso para nuestro propósito dentro del ámbito médico. Para mejorar la traducción se ha utilizado además como corpus de entrenamiento el vocabulario MeSH, que ha sido desarrollado manualmente por expertos humanos y proporciona una correspondencia muy fiable de miles de descriptores compuestos por sustantivos y adjetivos en múltiples pares de idiomas.[7]

Conviene clarificar que aunque la traducción automática no es un método con una precisión suficiente para sustituir las traducciones realizas por expertos, sí puede ser capaz de proporcionar buenos resultados para las expresiones textuales que utiliza el algoritmo. Un documento traducido puede no resultar muy coherente para una lectura fluida, pero una vez procesado para obtener los sintagmas nominales que forman las características, las expresiones son independientes las unas de los otras y, por lo tanto, no necesitan formar parte de un contexto específico y cohesionado.

6.1.5. Arquitectura del sistema

Para desarrollar el sistema de diagnóstico médico se ha empleado una arquitectura cliente servidor, con los siguientes requerimientos para cada una de las partes:

- En la parte del servidor, se requiere un ordenador preferiblemente con plataforma linux y multiprocesador, con unos 4GB de memoria RAM. No se requiere entorno gráfico.
- En la parte del cliente, no se requiere ningún hardware ni software especial para interactuar con el asistente. Simplemente se necesita conexión de Internet y un navegador estándar para acceder on-line a la web `www.lookfordiagnosis.com`.

Universitat d'Alacant
Universidad de Alicante

Usando una interfaz web y mostrando los resultados en formato de texto, se permite a los usuarios acceder desde diversos tipos de dispositivos portátiles (ordenadores, móviles, tabletas, etc.). Además, los usuarios siempre dispondrán de la última versión, pues no se requerirá instalar ninguna aplicación específica ni actualizaciones de software.

6.1.6. Ejemplos de funcionamiento

A continuación se muestran dos ejemplos de funcionamiento del sistema detallados paso a paso. En el primero de ellos utilizamos uno de los 400 historiales médicos incluidos en el conjunto de prueba, con las correspondientes salidas del algoritmo y la evaluación de los resultados. En el segundo ejemplo incluido en esta sección, y en los otros dos que podemos encontrar en el apéndice E, podemos observar historiales clínicos reales en español obtenidos

[7] Podemos encontrar por ejemplo traducciones del MeSH al español, francés, portugués, italiano, sueco, alemán, noruego, finlandés, ruso, polaco, rumano, griego, holandés, eslovaco, esloveno, turco, árabe, chino y japonés [Nelson et al., 2004].

Case 177 – Headaches, Lethargy and a Sellar/Suprasellar Mass

PATIENT HISTORY: A 16 year old female presented with two months of progressively worsening headaches, lethargy and visual disturbances. Her past medical history included developmental delay, shunted hydrocephalus, and tethered cord release. Neurological examination displayed no focal lesions, and endocrinologic studies were unremarkable.

IMAGING STUDIES: Magnetic resonance images (MRI) of the brain revealed an intrasellar and suprasellar cystic mass with well defined smooth borders, which displaced the optic chiasm superiorly. The lesion displayed high signal intensity on T1 weighted images (Images 1a and 1b, sagittal and axial, respectively) and showed peripheral enhancement with Gadolinium administration (Image 1c). The lesion also showed high signal intensity on T2 weighted images (Image 1d). A previous brain MRI from 1991 did not reveal sellar or suprasellar lesions.

PATHOLOGICAL FINDINGS: Intraoperative frozen sections showed cholesterol clefts and fragments of reactive squamous-appearing epithelium suggestive of a craniopharyngioma; however, a descriptive diagnosis was made in the absence of definitive histologic features. Routine histology utilizing hematoxylin and eosin staining revealed an epithelial-lined cyst with a thickened wall adjacent to a portion of benign adenohypophysis (Images 2a and 2b). The cyst lining consisted of one to two layers of ciliated cuboidal epithelium (Image 2c), with a focal simple squamous component. The thickened fibrous wall of the cyst displayed scattered foci of mixed acute and chronic inflammation with areas resembling granulation tissue. A small cluster of microscopic cysts lined by simple cuboidal epithelium was embedded in one portion of the wall. Another area of the cyst wall displayed prominent collections of cholesterol clefts associated with multinucleated giant cells, non-caseating granulomatous inflammation, hemosiderin- laden macrophages, disruption of the epithelial lining, and hemorrhage into the cyst lumen (Image 2d). Special stains for acid fast bacilli, fungi, and bacteria were negative.

Figura 6.4: Historial médico de paciente con quistes del sistema nervioso central. El texto completo del caso se puede obtener de http://path.upmc.edu/cases/case177.html.

de diversas páginas web, que aunque no han sido utilizados para la evaluación global del proceso de diagnóstico, sí muestran la posibilidad de utilizar síntomas especificados en otros idiomas diferentes del utilizado durante el proceso de aprendizaje.

Ejemplo 1: Quistes del sistema nervioso central

El diagnóstico proporcionado por los expertos para la historia clínica de la figura 6.4 es "Rathke's Cleft Cyst", que es un sinónimo del término "Central Nervous System Cysts". Transladando esto a una o varias de las 23 enfermedades del nivel más alto del MeSH, se obtienen las siguientes entradas:

- "Neoplasms"
- "Nervous System Diseases"
- "Congenital, Hereditary, and Neonatal Diseases and Abnormalities"

La figura 6.5 muestra la salida del sistema para la anterior historia clínica, obteniendo los siguientes valores tras su evaluación:

- $P_{r=0} = 1$, puesto que el primer elemento ("Neoplasms") coincide con uno de los diagnósticos dado por los expertos
- $AvgP = \frac{\frac{1}{1} + \frac{2}{2} + \frac{3}{4}}{3} = 0.92$, puesto que los diagnósticos correctos se hayan en las posiciones 1, 2 y 4
- $BEP = \frac{2}{3} = 0.67$

Ejemplo 2: Angina inestable

El diagnóstico proporcionado por los expertos para la historia clínica de la figura 6.6 es "Angina inestable". Transladando esto a una o varias de las 23 enfermedades del nivel más alto del MeSH, se obtienen las siguientes entradas:

- "Enfermedades Cardiovasculares"
- "Condiciones Patológicas, Signos y Síntomas"

Universitat d'Alacant
Universidad de Alicante

La figura 6.7 muestra la salida del sistema para la anterior historia clínica, obteniendo los siguientes valores tras su evaluación:

- $P_{r=0} = 1$, puesto que el primer elemento ("Condiciones Patológicas, Signos y Síntomas") coincide con uno de los diagnósticos dados por los expertos
- $AvgP = \frac{\frac{1}{1} + \frac{2}{2}}{2} = 1$, puesto que los diagnósticos correctos se hayan en las posiciones 1 y 2
- $BEP = \frac{2}{2} = 1$

Symptoms:

Patient history: a 16 year old female presented with two months of progressively worsening headaches, lethargy and visual disturbances. her

Check | Medicinal plants

Fast | Hierarchical

More information:

3t3 cells; bacteria; brain; cells; cholesterol; craniopharyngioma; cysts; diagnosis; epithelial cells; epithelium; frozen sections; fungi; gadolinium; giant cells; granulation tissue; hematoxylin; hemorrhage; histology; history; hydrocephalus; image enhancement; inflammation; lethargy; lethargy; macrophages; magnetic resonance imaging; negative staining; optic chiasm

Order	Diseases	Rating	Bibliogr.	Expand
1	Neoplasms	100	+ + +	->
2	Nervous System Diseases	76	+ + +	->
3	Pathological Conditions, Signs and Symptoms	71	+ + +	->
4	Congenital, Hereditary, and Neonatal Diseases and Abnormalities	28	+ + +	->
5	Musculoskeletal Diseases	23	+ + +	->
6	Nutritional and Metabolic Diseases	20	+ + +	->
7	Cardiovascular Diseases	19	+ + +	->
8	Bacterial Infections and Mycoses	17	+ + +	->
9	Eye Diseases	16	+ + +	->
10	Skin and Connective Tissue Diseases	16	+ + +	->
11	Hemic and Lymphatic Diseases	16	+ + +	->
12	Digestive System Diseases	15	+ + +	->
13	Endocrine System Diseases	14	+ + +	->
14	Disorders of Environmental Origin	13	+ + +	->
15	Stomatognathic Diseases	13	+ + +	->
16	Respiratory Tract Diseases	13	+ + +	->
17	Female Urogenital Diseases and Pregnancy Complications	13	+ + +	->
18	Immune System Diseases	11	+ + +	->
19	Male Urogenital Diseases	8	+ + +	->
20	Mental Disorders	6	+ + +	->
21	Otorhinolaryngologic Diseases	6	+ + +	->
22	Virus Diseases	4	+ + +	->
23	Parasitic Diseases	3	+ + +	->

Universitat d'Alacant
Universidad de Alicante

Figura 6.5: Ejemplo del primer nivel de diagnóstico jerárquico para un caso de quistes del sistema nervioso central (correspondiente al historial de la figura 6.4).

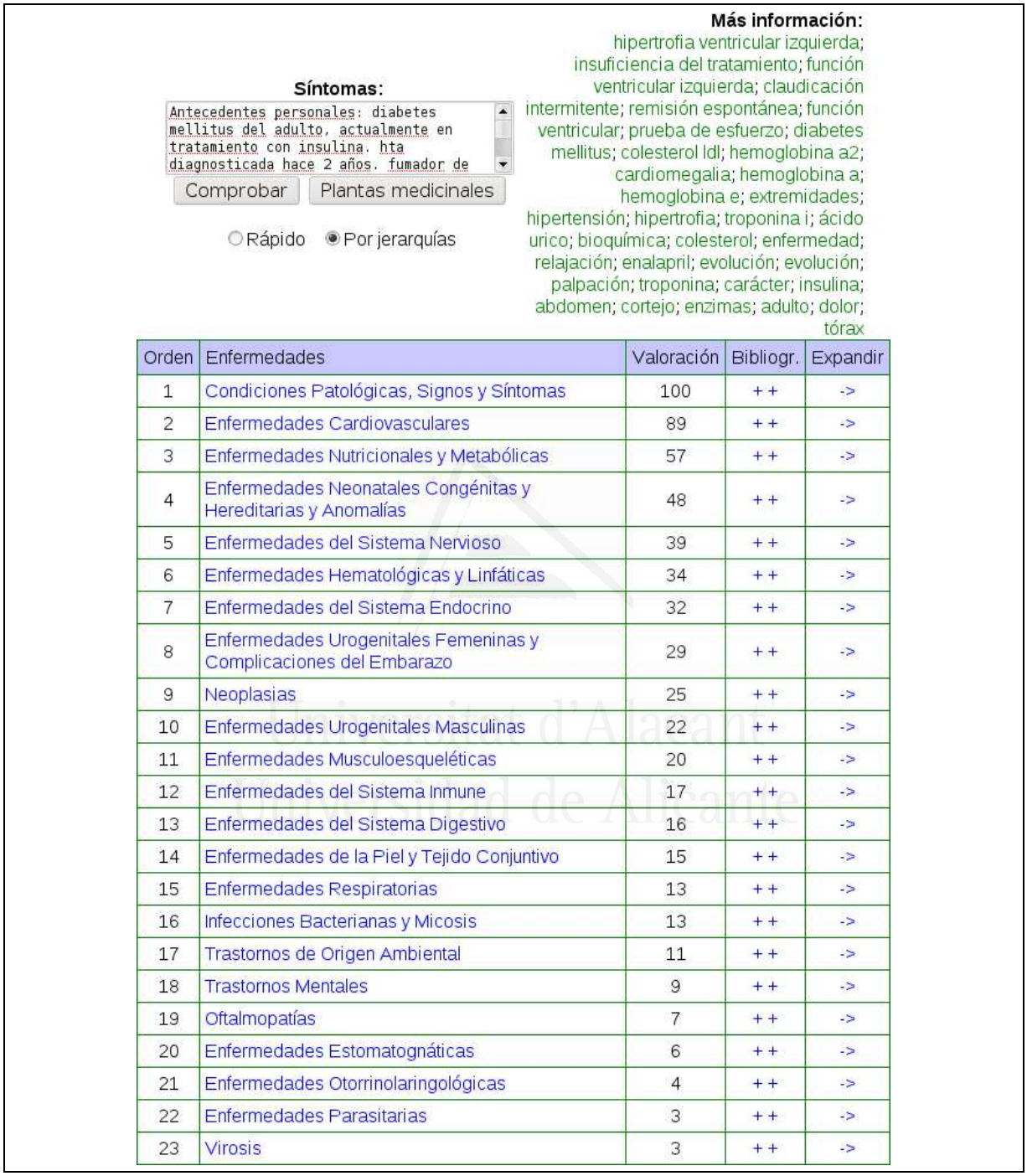
Antecedentes personales: Diabetes Mellitus del adulto, actualmente en tratamiento con insulina. HTA diagnosticada hace 2 años. Fumador de menos de 10 cigarrillos diarios hasta hace 3 meses. Claudicación intermitente. Tratamiento habitual: insulina NPH (20-0-18), enalapril 20 mg (1 comp/dia) y ácido acetilsalicílico (100 mg/día).

Enfermedad actual: paciente de 73 años, que ingresa por haber presentado dolor torácico, de instauración paulatina tras realizar esfuerzo moderado, localizado en región centrotorácica, de carácter constrictivo, sin irradiación ni cortejo vegetativo, de 1 hora de duración, en relación con sus actividades agrícolas, con remisión espontánea. Al día siguiente ha presentado otros dos más, de menor duración y mayor intensidad, en uno de ellos con agravamiento de su hipertensión, motivo por el que se traslada a urgencias para ingreso.

Exploración: TA: 130/80. No refiere dolor torácico. Temperatura: 36.9 . Consciente y orientado temporoespacialmente. Eupneico. Bien hidratado y perfundido. No adenopatías palpables en región cervical, supraclavicular, axilar o inguinal. No estasis yugular. AC. Tonos rítmicos a 55/min, sin auscultarse soplos. AR. Murmullo vesicular conservado. Abdomen: blando y depresible, no doloroso a palpación. No organomegalias palpables. Extremidades sin alteraciones significativas, salvo disminución de ambos pulsos pedios.

Analítica: hemograma con hemoglobina: 13.5 gr/dl, HTO: 49 %, hem: 4960000. Bioquímica al ingreso: glucemia: 143 mg/dl, colesterol total: 205 mg/dl, LDL: 132 mg/dl, ácido úrico: 8.7 mg/dl, troponina I: 2.49 ng/ml (0-0.1), GOT: 35 UI/L (2-34). CK, CK-MB y LDH normales. Coagulación básica (TP y TPTA) normales. En la bioquímica posterior (siguientes 7 días), únicamente se registra una troponina I de 0.49 ng/ml a los tres días, normalizandose posteriormente.Resto de los enzimas normales (CK, CK-MB, LDH, GOT) Rx. torax: discreta cardiomegalia a expensas del ventrículo izquierdo. Ecocardiograma: válvulas con signos degenerativos e insuficiencia mitral mínima. Hipertrofia ventricular izquierda concéntrica (13-14 mm) sin defectos de contractilidad y función sistólica normal (FE: 60 %). Trastorno de la relajación en el flujo mitral. Compatible con cardiopatía hipertensiva. ECG: RS a 60/min. Rectificación no significativa del ST en V5 y V6. Evolución: desde que el paciente acudió a urgencias y durante el ingreso el paciente ha permanecido asintomático, no volviendo a presentar dolor torácico en reposo ni deambulando por el servicio. Sin cambios en los electrocardiogramas de control. Existe divergencia sobre la prueba/s diagnóstica(s) a realizar.

Figura 6.6: Historial médico de paciente con angina inestable. El texto completo del caso se puede obtener de http://casosclinicos.webcindario.com/rccsep2003.htm.



Orden	Enfermedades	Valoración	Bibliogr.	Expandir
1	Condiciones Patológicas, Signos y Síntomas	100	+ +	->
2	Enfermedades Cardiovasculares	89	+ +	->
3	Enfermedades Nutricionales y Metabólicas	57	+ +	->
4	Enfermedades Neonatales Congénitas y Hereditarias y Anomalías	48	+ +	->
5	Enfermedades del Sistema Nervioso	39	+ +	->
6	Enfermedades Hematológicas y Linfáticas	34	+ +	->
7	Enfermedades del Sistema Endocrino	32	+ +	->
8	Enfermedades Urogenitales Femeninas y Complicaciones del Embarazo	29	+ +	->
9	Neoplasias	25	+ +	->
10	Enfermedades Urogenitales Masculinas	22	+ +	->
11	Enfermedades Musculoesqueléticas	20	+ +	->
12	Enfermedades del Sistema Inmune	17	+ +	->
13	Enfermedades del Sistema Digestivo	16	+ +	->
14	Enfermedades de la Piel y Tejido Conjuntivo	15	+ +	->
15	Enfermedades Respiratorias	13	+ +	->
16	Infecciones Bacterianas y Micosis	13	+ +	->
17	Trastornos de Origen Ambiental	11	+ +	->
18	Trastornos Mentales	9	+ +	->
19	Oftalmopatías	7	+ +	->
20	Enfermedades Estomatognáticas	6	+ +	->
21	Enfermedades Otorrinolaringológicas	4	+ +	->
22	Enfermedades Parasitarias	3	+ +	->
23	Virosis	3	+ +	->

Figura 6.7: Ejemplo del primer nivel de diagnóstico jerárquico para un caso de angina inestable (correspondiente al historial de la figura 6.6, página 150).

Corpus	Algoritmo	$P_{r=0}$	AvgP	BEP
OHSUMED	SVM	-	-	0.66
	Suma de Pesos	-	-	0.71
Casos registrados e historiales de pacientes	Suma de Pesos	0.69	0.73	0.62

Tabla 6.1: Resultados comparativos para los procesos de categorización sobre OHSUMED (23 categorías) y diagnóstico de enfermedades (400 historiales médicos clasificados sobre 23 categorías). Los valores se muestran en una misma tabla para poder apreciar la similitud de rendimiento obtenida sobre ambos corpus.

6.1.7. Resultados

Para evaluar la validez de los diagnósticos proporcionados automáticamente procederemos a clasificar un conjunto de documentos formado por los síntomas de 400 pacientes, y observaremos la calidad del ranking de enfermedades obtenido en cada caso, calculando finalmente la media para tener una visión objetiva de los resultados generales. De cada documento hemos extraído el título y la historia clínica, que constituyen el texto proporcionado al algoritmo para decidir la relevancia de cada una de las categorías. También disponemos de los diagnósticos ofrecidos por expertos humanos para poder comparar entre los resultados obtenidos y los esperados.

La última fila de la tabla 6.1 muestra los resultados obtenidos para la clasificación de los 400 historiales médicos, promediando la valoración de las 400 decisiones en un solo valor. $P_{r=0}$ indica que un 69 % de los historiales obtuvieron el diagnóstico correcto con la entrada del MeSH que se ofreció en primera posición. La medida AvgP indica que el resto de la lista también contiene valores válidos pues alcanza un valor de 73 %.

A modo de comparación hemos incluido en la primera fila de la tabla 6.1 los resultados obtenidos por SVM [Joachims, 1998] y nuestro sistema basado en suma de pesos utilizando el corpus OHSUMED. Puesto que estamos trabajando con corpus diferentes, no podemos realizar comparaciones de manera directa. Sin embargo, sí podemos apreciar que el valor de BEP obtenido para el diagnóstico médico sobre los 400 expedientes es equiparable a los alcanzados en tareas convencionales de categorización de textos sobre OSHUMED. Esto lleva a la conclusión de que el ranking de categorías podría resultar útil aplicándolo en el ámbito del diagnóstico médico.

6.2. Otras aplicaciones

A continuación hablaremos de otras aplicaciones del algoritmo de clasificación de textos descrito en los capítulos anteriores. Aunque en esta sección no llevaremos a cabo evaluaciones de los sistemas propuestos, sí mostraremos versiones completamente operativas de los mismos. Describiremos, por ejemplo, cómo obtener medicamentos o plantas medicinales a partir de ciertos síntomas, o cómo construir un directorio médico partiendo de los artículos publicados

en la Wikipedia.

Para todas las propuestas, abordaremos el tema de la multilingüalidad de la misma forma que en la sección anterior, permitiendo realizar consultas en gran cantidad de idiomas. Sólo requeriremos traducir las palabras clave en cuestión al inglés para que el clasificador pueda procesarlas como entrada, seleccionando a su vez también el correspondiente idioma del MeSH para obtener el ranking de categorías que se muestra al usuario.

6.2.1. Prescripción de medicamentos

Del mismo modo que se pueden ordenar las categorías dentro del sub-árbol de enfermedades, utilizando otras partes del árbol MeSH se puede conseguir ordenar una lista de medicamentos en función de los síntomas introducidos. Para ello, podemos utilizar el listado internacional de principios activos (sustancia principal en la que se basa la acción terapéutica de cada medicamento). En la página www.vademecum.es[8] se pueden consultar los principios activos de la mayoría de los medicamentos comercializados en España.

Partiendo del ejemplo mostrado en la figura 6.8, podemos observar cómo obtener los principios activos relacionados con ciertos síntomas mediante clasificación automática. En concreto, para el "dolor de cabeza" se puede observar que en la primera posición se recomienda "Dihidroergotamina", que se utiliza como profilaxis de la jaqueca y equivalentes jaquecosos.[9] En la página web www.lookfordiagnosis.com se pueden realizar pruebas reales de dicha aplicación a través de la opción "Principios activos".

Para entrenar el algoritmo para este nuevo propósito, hemos utilizado una consulta sobre la base de datos PubMed que nos devuelve todos los documentos relacionados con plantas medicinales o principios activos, que hablan de sus efectos terapéuticos y no de los efectos secundarios que puedan tener[10]. Después de filtrar los artículos escritos en inglés que contienen resumen, se obtienen un total de 540.235[11] documentos adecuados para la fase de entrenamiento. Como salida del proceso de clasificación se obtendrá una lista ordenada de 7.994 principios activos.

Para observar la capacidad multilingüe, en la figura 6.9 se muestra un ejemplo en inglés con los principios activos recomendados para "rheumatoid arthritis", junto con el enlace a uno de los medicamentos que contienen el primer principio activo mostrado en la lista. Como se puede observar, en primera posición se recomienda "Gold", que desde el enlace proporcionado en la columna "Drugs" nos lleva al medicamento "Myocrisin", que contiene dicho principio activo.[12]

[8] http://www.vademecum.es/principios-activos-a_1.

[9] http://www.vademecum.es/principios-activos-DIHIDROERGOTAMINA-N02CA01.

[10] Se puede observar el texto completo de la consulta en la figura F.1, página 197.

[11] Cifra obtenida en febrero de 2007. En mayo de 2013 obtenemos 730.808 documentos con la misma consulta.

[12] Más información en http://www.netdoctor.co.uk/medicines/100001780.html.

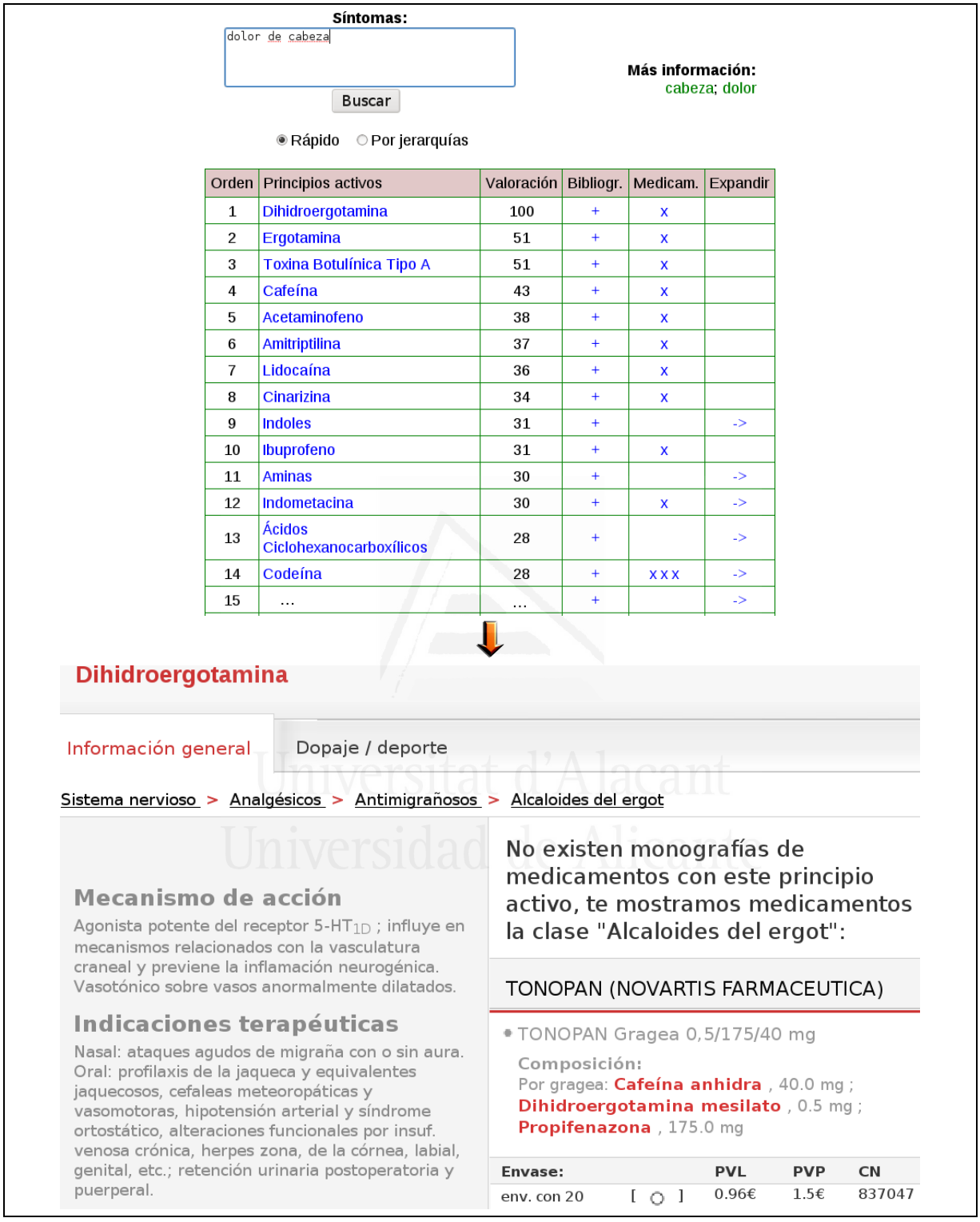


Figura 6.8: Ejemplo de sugerencia en la prescripción para *dolor de cabeza* (arriba) y el nombre del medicamento (abajo) encontrado mediante el enlace proporcionado por nuestro sistema. Las cifras especificadas en la columna "Valoración" se refieren a la ponderación que da el algoritmo a cada una de las opciones, indicando un valor de confianza. La primera opción siempre tendrá el valor máximo (100) y el resto de principios activos se ordenan de manera descendente.

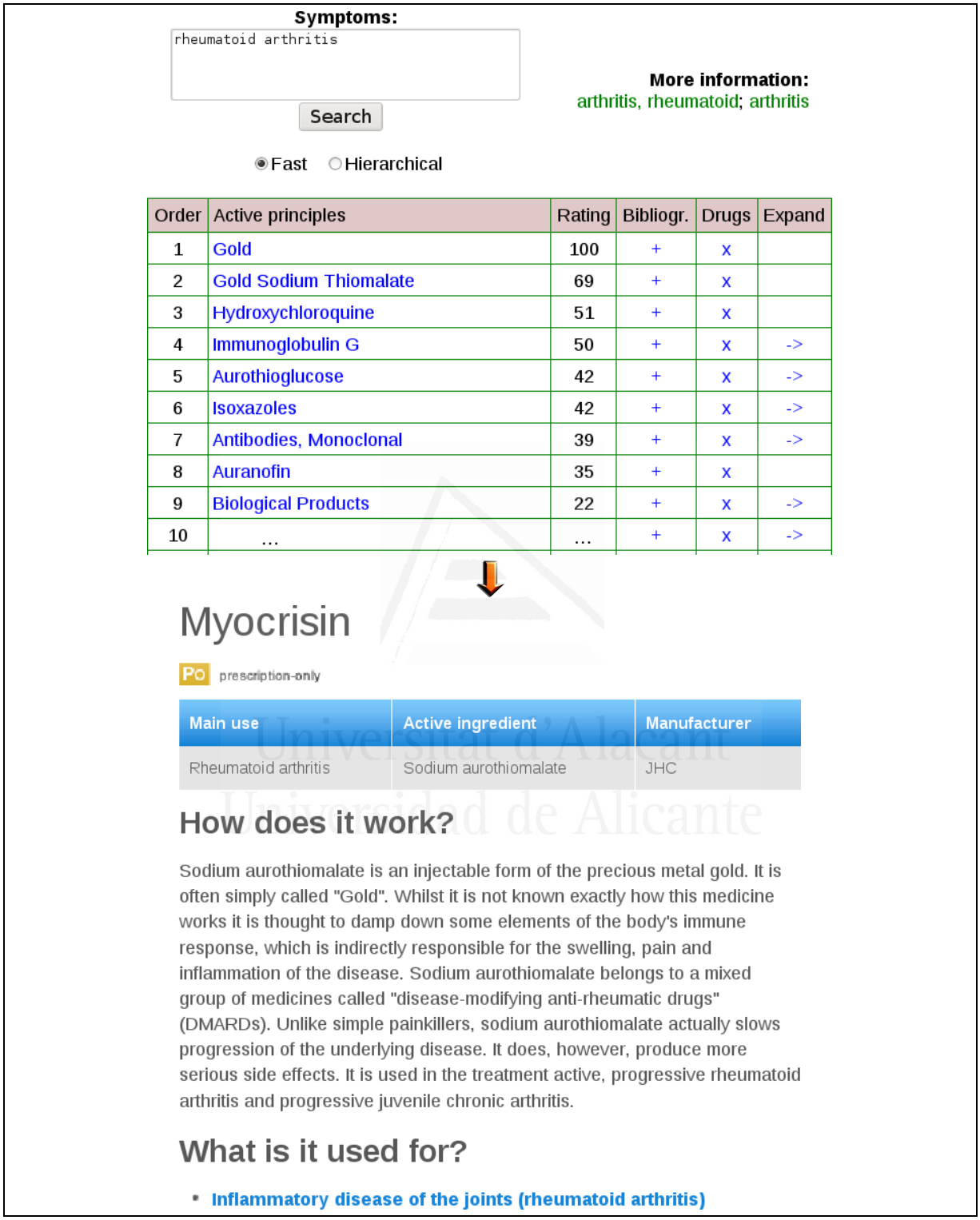


Figura 6.9: Ejemplo de sugerencia en la prescripción para *rheumatoid arthritis* (arriba) y el nombre del medicamento (abajo) encontrado mediante el enlace proporcionado por el asistente.

Todavía no se ha comprobado la efectividad de las prescripciones comparando con datos ofrecidos por expertos. Y aunque la salida del clasificador debería ser supervisada siempre por un médico, sí podría utilizarse como asistente. Por ello, esfuerzos futuros se dirigirán en realizar diversas pruebas para intentar evaluar la fiabilidad de los resultados.

6.2.2. Selección de plantas medicinales

Se trata de un caso particular del expuesto en la sección anterior. Puesto que se dedican menos esfuerzos a la investigación sobre plantas medicinales, para poder conseguir suficientes muestras para el aprendizaje se debe de ampliar la búsqueda de documentos de entrenamiento. Por ello, hemos cambiado el sufijo "majr" por "mh" para indicar que los documentos a extraer de la base de datos, aunque deben tratar sobre plantas medicinales, no tienen porqué versar principalmente de éste tema. Además eliminamos la restricciones de que los artículos estén relacionados con humanos o que se trate de casos registrados.

De esta forma, para entrenar el algoritmo para este nuevo propósito, hemos utilizado una consulta sobre la base de datos PubMed que nos devuelve los artículos relacionados con plantas medicinales que no traten sobre sus efectos secundarios[13]. Después de filtrar los artículos escritos en inglés que contienen resumen, se obtienen un total de 113.649[14] documentos adecuados para la fase de entrenamiento. Como salida del proceso de clasificación se obtendrá una lista ordenada de 1.117 plantas medicinales.

En la figura 6.10 se puede observar un ejemplo de cómo se llega a un listado ordenado de plantas medicinales partiendo únicamente de la palabra clave "insomnio". En la página web www.lookfordiagnosis.com se pueden realizar pruebas reales de dicha aplicación, accediendo desde la página principal, a través del botón "Plantas medicinales". Algunas comprobaciones manuales demuestran la validez de los resultados proporcionados en dicho ejemplo (valeriana, lavandula, cucurbita y kava).

Sin embargo, todavía no se ha comprobado la efectividad de los resultados proporcionados comparando con datos ofrecidos por expertos. Por ello, esfuerzos futuros también se dirigirán a realizar pruebas exhaustivas para intentar evaluar la fiabilidad de los resultados.

6.2.3. Ranking de temas relacionados en búsquedas médicas

Resulta bastante común en los buscadores obtener un listado de temas relacionados con las palabras clave introducidas por el usuario. Los buscadores médicos no son ajenos a esto. En la figura 6.11 se puede observar el resultado ofrecido por un conocido buscador médico (searchmedica.com) al buscar información sobre alergias ("allergies"). Como se lee en la

[13] Se puede observar el texto completo de la consulta en la figura F.2, página 197.

[14] Cifra obtenida en febrero de 2007. En mayo de 2013 obtenemos 316.720 documentos con la misma consulta.

Orden	Plantas medicinales	Valoración	Bibliogr.	Expandir
1	Valeriana	100	+ +	
2	Lavandula	22	+ +	
3	Cucurbita	16	+ +	
4	Kava	14	+ +	

Figura 6.10: Ejemplo de sugerencia de plantas medicinales para "insomnio".

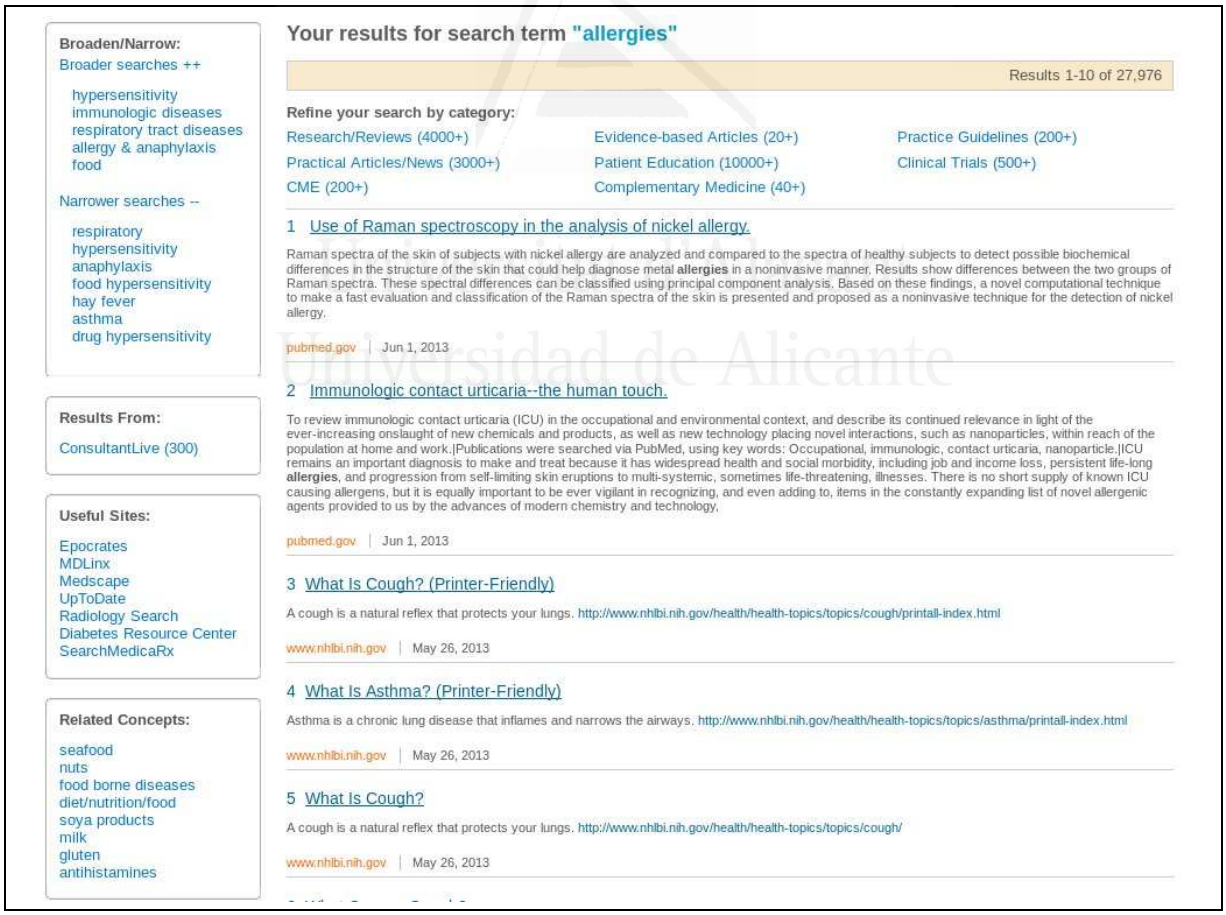


Figura 6.11: Ejemplo de búsqueda de información sobre "allergies" en `searchmedica.com`.

parte izquierda, en los cuadros de ampliar o reducir la búsqueda ("Broaden/Narrow") y conceptos relacionados ("Related Concepts"), se obtienen las siguientes expresiones:

- *Broader searches: hypersensitivity, immunologic diseases, respiratory tract diseases, allergy & anaphylaxis, food*
- *Narrower searches: respiratory hypersensitivity, anaphylaxis, food hypersensitivity, hay fever, asthma, drug hypersensitivity*
- *Related Concepts: seafood, nuts, food borne diseases, diet/nutrition/food, soya products, milk, gluten, antihistamines*

Como se puede observar, no se muestra por ejemplo ninguna referencia a términos tales como "rhinitis", "dermatitis", "eczema", "pollen" o "insect bites", que a lo largo de los años han resultado ser términos muy frecuentes en consultas realizadas en la mayoría de los motores de búsqueda.[15]

En este apartado describimos nuestra propuesta para la obtención de temas relacionados con búsquedas médicas. Para el ejemplo anterior sobre alergias, algunas de las palabras clave que incluiría nuestro sistema son:

Food Hypersensitivity, Hypersensitivity, Allergy and Immunology, Drug Hypersensitivity, Latex Hypersensitivity, Milk Hypersensitivity, Desensitization-Immunologic, Allergens, Hypersensitivity-Immediate, Skin Tests, Anaphylaxis, Rhinitis-Allergic-Seasonal, Dermatitis-Occupational, Dermatitis-Allergic Contact, Conscience, Arachis hypogaea, Rhinitis-Allergic-Perennial, Eczema, Pollen, Insect Bites and Stings, Egg Hypersensitivity, Latex, Asthma, Rhinitis, Bee Venoms, Respiratory Hypersensitivity, Gloves-Surgical, Anisakis, Conjunctivitis-Allergic, Urticaria, ...

En la figura 6.12 se puede observar el resultado obtenido directamente de la página web www.lookformedical.com para la consulta "allergies". En la parte izquierda se muestran algunas de las 25.000 entradas del MeSH clasificadas según orden de relevancia para la palabra clave "allergies".

Los documentos de aprendizaje se han obtenido también de la base de datos PubMed[16]. Pretendemos extraer un conjunto de muestras formado por todos los documentos clasificados dentro del árbol MeSH, lo que nos permitirá disponer de gran cantidad de textos de entrenamiento que nos ayuden a decidir posteriormente si ciertas palabras clave están relacionadas con cualquier categoría. Después de limitar la búsqueda a sólo aquellos artículos escritos en inglés con resumen y con enlace gratuito al texto completo (*Links to free full text*[17]) se obtienen un total de 1.869.136 documentos. Debido a la gran cantidad de categorías del MeSH

[15] Se puede consultar por ejemplo Google Trends (http://www.google.com/trends/) para analizar la relevancia de dichos términos.

[16] Se puede observar el texto completo de la consulta propuesta en la figura F.3, página 197.

[17] Debido a la gran cantidad de textos de entrenamiento de la que disponemos, podemos filtrar aquellos artículos médicos que no proporcionan un enlace al texto completo de la publicación de forma gratuita. Posteriormente podemos proporcionar dichos enlaces al usuario para ampliar el resultado de su búsqueda.

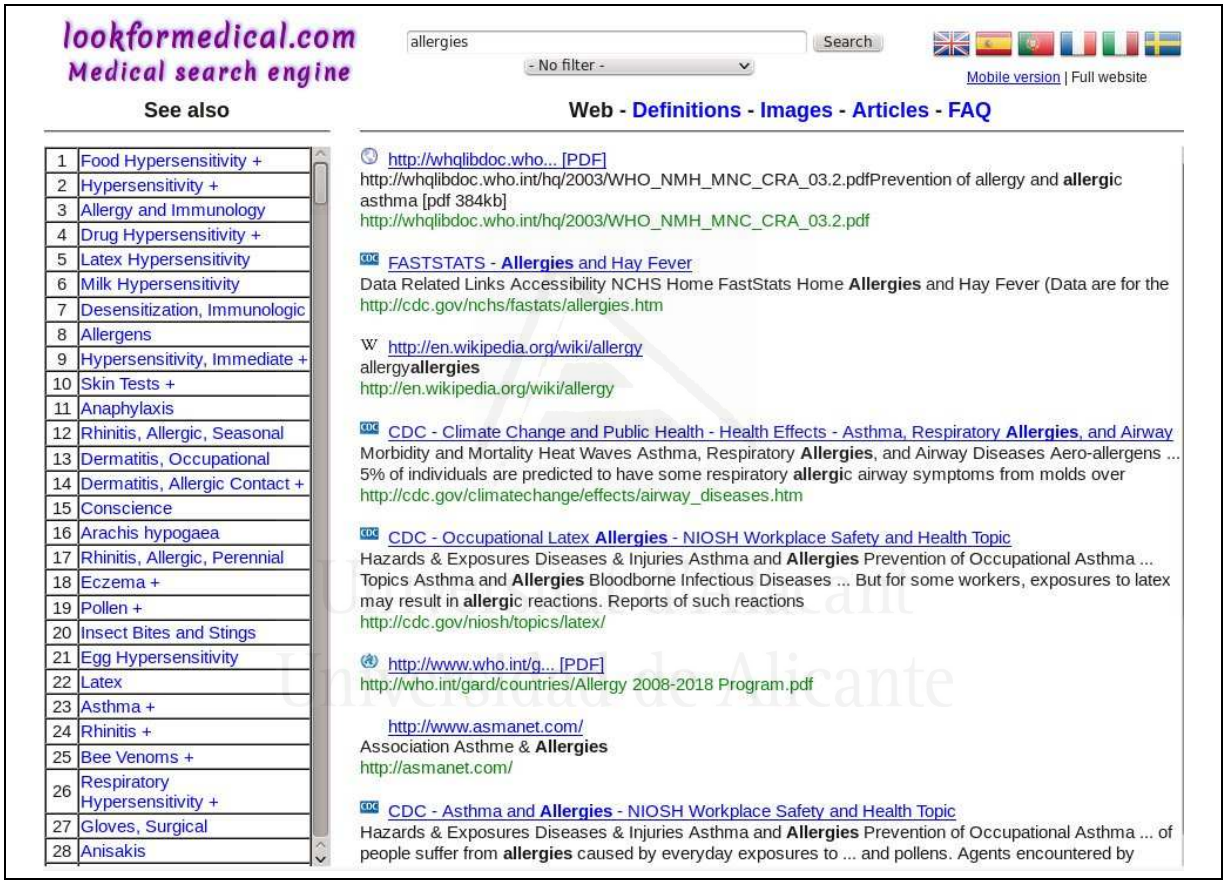


Figura 6.12: Ejemplo de búsqueda de información sobre "allergies" en `lookformedical.com`.

(casi 25.000) y la gran cantidad de documentos (casi 2 millones), no todos los algoritmos permiten el manejo del árbol completo [Ibushi y Tsujii, 1999; Yang et al., 2003]. Al igual que en el resto de aplicaciones planteadas, nosotros proponemos utilizar nuestro sistema basado en la suma de pesos por su simplicidad y eficiencia.

Una vez procesados todos los datos de entrada y finalizado el proceso de aprendizaje, las pruebas demuestran que con unos recursos computacionales estándar,[18] el algoritmo es capaz de completar el entrenamiento en pocas horas, pudiendo clasificar nuevos documentos con un tiempo de respuesta casi inapreciable. Es decir, puede funcionar con velocidad suficiente para ser utilizado en un buscador, incrementando mínimamente el tiempo de respuesta.

6.2.4. Directorio médico

Algunos directorios web han incrementado su popularidad en los últimos años. Es el caso, por ejemplo, del "Open Directory Project" (ODP),[19] mantenido manualmente por una amplia comunidad de editores. En la actualidad proporciona enlaces a más de cinco millones de páginas web que se encuentran organizadas en más de un millón de categorías estructuradas jerárquicamente. Este tesauro se utiliza por otras empresas como Alexa,[20] que permiten obtener un listado de páginas web ordenadas por cantidad de visitas en el ámbito de ciertas temáticas, seleccionando las categorías de más a menos genéricas.

Wikipedia se encuentra indexada por Alexa en la sexta posición a nivel mundial. Debido a la gran cantidad de información y la gran difusión de los temas que se tratan en la Wikipedia, resulta difícil clasificar automáticamente sus contenidos en un árbol estándar, tales como los empleados en los directorios web. Tal como se realiza en ODP, actualmente esta tarea se suele realizar a mano por un numeroso grupo de voluntarios.

Nuestra propuesta parte por intentar identificar automáticamente los artículos médicos contenidos en la Wikipedia, para proporcionar acceso a cada uno de ellos a través de un directorio web organizado por categorías. Con este propósito hemos empleado nuestro sistema para filtrar los artículos por temas. En la figura 6.13 podemos observar un ejemplo de funcionamiento a través de la página web www.lookforwiki.com. Como se muestra en la parte izquierda, el usuario selecciona un idioma y automáticamente obtiene una lista de temas seguido por el número de artículos de la Wikipedia relacionados con dicho tema. Se puede acceder a temas más específicos navegando a lo largo del árbol. El usuario puede obtener finalmente los enlaces a los artículos seleccionando el nombre de la categoría.

[18] Las pruebas se han realizado con 8 GB de RAM, y un procesador con cuádruple núcleo a 2,66 Ghz.
[19] http://www.dmoz.org/docs/en/about.html.
[20] http://www.alexa.com/company.

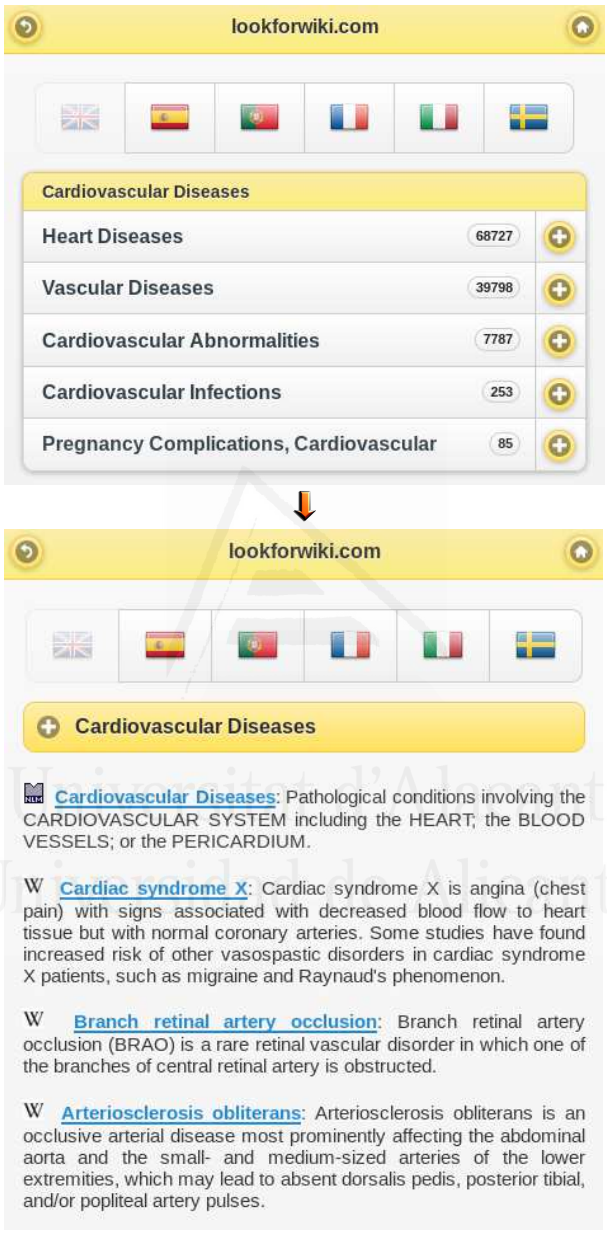


Figura 6.13: Ejemplo de recorrido del directorio web para localizar artículos relacionados con "Cardiovascular Diseases". El número de artículos disponibles se muestra en la imagen superior, y la lista con los enlaces se observa en la parte inferior.

Universitat d'Alacant
Universidad de Alicante

Capítulo 7

Conclusiones y trabajo futuro

El objetivo de esta tesis ha sido la construcción de un sistema de clasificación de textos adaptable a diversos dominios y que pueda cumplir con los requisitos básicos para ser utilizado en aplicaciones del mundo real. Dentro de todos los posibles ámbitos, nos hemos decantado por su uso para el desarrollo de asistentes médicos. Debido a la alta disponibilidad de publicaciones científicas, tenemos a nuestro alcance una enorme cantidad de documentos para llevar a cabo la fase de entrenamiento. Por ello hemos centrado nuestra investigación dentro del aprendizaje automático supervisado, intentando ante todo que el proceso de aprendizaje sea completamente autónomo y que no requiera ajustes manuales ni intervención humana.

Dentro del creciente mundo de los motores de búsqueda y las bibliotecas digitales, todavía encontramos desde nuestro punto de vista la necesidad de poder limitar las búsquedas por ramas temáticas. Tal es el propósito y a ello se debe la existencia de algunos directorios web que son mantenidos manualmente. Nuestro propósito es seguir esos pasos para conseguir categorizar automáticamente cualquier página web para captar y listar aquellas que puedan tener algún tipo de utilidad dentro del ámbito médico. De esta forma lograremos acotar o ampliar búsquedas no sólo por palabras claves, sino por conceptos o categorías.

En el futuro pretendemos llegar a tantos usuarios como sea posible, incrustando las aplicaciones propuestas en entornos multilingües que puedan ser ejecutados en gran cantidad de dispositivos, tanto fijos como móviles. Pero no sólo focalizaremos nuestros esfuerzos en intentar mejorar la accesibilidad y facilidad de uso: nuestro objetivo principal y final será el desarrollo de aplicaciones mantenidas de forma automática, con un tiempo de respuesta reducido y con un creciente aumento en la precisión de los resultados.

7.1. Conclusiones

Cuando se trabaja con clasificación de textos, resulta decisivo utilizar un modelo adecuado de selección y ponderación de características. En esta tesis se ha propuesto una estrategia

para escoger, extraer y ponderar características de documentos en texto plano de forma que se obtenga una representación más eficiente que con métodos tradicionales, tales como la bolsa de palabras.

Además, el algoritmo que hemos planteado es eficiente desde el punto de vista computacional, fácil de implementar, se adapta a cada dominio de manera automática y utiliza un proceso de aprendizaje incremental (sólo los nuevos documentos deben ser procesados después de actualizar el conjunto de aprendizaje).

También se ha demostrado que la suma de los pesos calculados mediante el algoritmo propuesto es suficiente para clasificar nuevos documentos, obteniendo un rendimiento mayor que otros algoritmos de clasificación más complejos.

A grandes rasgos, con el nuevo modelo se consigue:

- Independencia del corpus: al estar basado en información puramente estadística, y no lingüística, el sistema es fácilmente adaptable a diferentes idiomas[1] y dominios.
- Utilización únicamente de métodos estadísticos: el modelo cumple totalmente este requisito, pues los pesos de los términos se calculan a partir de datos extraíbles directamente de los textos sin ningún tipo de conocimiento lingüístico. Por ejemplo, aunque la mayoría de implementaciones utilizan las mencionadas listas de "palabras de parada" que suponen el uso de conocimiento lingüístico, no es este el caso de nuestra aproximación, puesto que dicha lista se obtiene automáticamente.
- Escalabilidad: esta ventaja se deriva del hecho de que es posible disponer de colecciones que crezcan de manera continua. Se puede actualizar el clasificador mediante nuevas muestras que incrementen su capacidad de clasificación sin tener que volver a procesar todos los documentos de entrenamiento. Por ejemplo, para calcular el peso de un término no es necesario conocer el número total de documentos en los que aparece dicho término. Por ello, la fase de indexado en la que se generan los vectores de los documentos de la colección, se puede realizar gradualmente a medida que se vayan añadiendo nuevos textos.

Respecto a las posibles aplicaciones del algoritmo, hemos demostrado que puede resultar efectivo en la asistencia de diagnósticos médicos. Hemos llevado a cabo un proceso de evaluación que incluye cientos de historiales reales de pacientes, observando los resultados obtenidos por nuestro sistema para poder obtener conclusiones respecto a la validez de las respuestas proporcionadas. Aunque, por motivos obvios, la salida del proceso de categorización no debe ser tomada como un diagnóstico final, consideramos que la precisión conseguida podría ser suficiente para asistir a expertos humanos. El sistema no se ha diseñado para que los propios pacientes realicen un autodiagnóstico, sino para que médicos especialistas puedan recibir diversas sugerencias de posibles enfermedades, llegando a la elección correcta en función de

[1] En el caso de idiomas tales como el chino o el japonés, harían falta herramientas específicas de tokenización.

sus propios criterios. Además, al proporcionar enlaces a los artículos médicos que llevan a la obtención de cada una de las categorías, se puede tener un rápido vínculo con la bibliografía que llevaría a las mismas conclusiones utilizando un proceso manual.

El potencial del sistema de clasificación propuesto se puede extender a otras posibles aplicaciones. Aunque todavía no hemos obtenido conclusiones empíricas en otros ámbitos diferentes al del diagnóstico médico, pensamos que el algoritmo planteado también puede ser utilizado para ofrecer sugerencias de medicamentos o de plantas medicinales. Además, hemos comprobado que la sencillez en el proceso de entrenamiento y la rapidez durante la fase de clasificación de nuevos documentos, ofrecen la posibilidad de emplear la categorización de textos para extender la funcionalidad de motores de búsqueda para el filtrado de páginas web, o la creación de directorios que puedan ser actualizados de forma automática. Esto facilita la navegación jerárquica para la localización de páginas web que versen sobre temas específicos, lo que permite reducir la cantidad de enlaces relevantes, o incluso obtener resultados adicionales de conceptos similares que posiblemente no se obtendrían al realizar una búsqueda iniciada mediante palabras clave.

7.2. Trabajo futuro

Tenemos previsto realizar una evaluación exhaustiva de todas las aplicaciones propuestas en el capítulo anterior. Pretendemos realizar un estudio recurriendo a usuarios de diversa índole, tales como estudiantes de medicina o expertos facultativos. Esto nos permitirá evaluar la efectividad de las aplicaciones desde un punto de vista tanto cuantitavio (rendimiento del algoritmo de clasificación) como cualitativo (grado de satisfacción), permitiendo poder comparar los resultados en función del perfil del usuario final. Además, puede resultar muy interesante recabar información específica de diferentes especialistas médicos (oncólogos, pediatras, traumatólogos, cardiólogos, ginecólogos, urólogos, etc.) de forma que podamos llevar a cabo también una evaluación jerárquica, partiendo de un uso genérico (desde medicina general por ejemplo) para ir descendiendo a cada una de las posibles ramas de la salud o ámbitos de aplicación (medicina alternativa, empresas farmacéuticas, centros de salud, hospitales, etc).

La incesante actualización de las bases de datos médicas conlleva la necesidad de un proceso de automatización de la fase de entrenamiento del clasificador para su puesta al día. Si además se pretende utilizar dicho sistema como asistente para el diagnóstico médico o para la prescripción de medicamentos, este requerimiento resulta ineludible. Con las propuestas planteadas, todavía no existe un vínculo directo entre la base de datos PubMed y el algoritmo de clasificación y, por ello, hoy por hoy no se puede llevar a cabo un aprendizaje que se active periódicamente de forma completamente autónoma. Por ejemplo, nuevos casos registrados de cada enfermedad surgen cada día, descubriéndose nuevas enfermedades y medicamentos que no aparecen en la interfaz web a menos que se realice una actualización mediante intervención humana. Puesto que disponemos de las herramientas necesarias, nuestro propósito es llevar a cabo dicho proceso de forma desatendida y que el resultado se refleje en la obtención de sugerencias de diagnósticos y principios activos recomendados de forma más precisa y

actualizados conforme vayan surgiendo nuevas publicaciones científicas.

Dentro del ámbito médico existe una notable escasez de herramientas especializadas en la búsqueda de páginas web. Podría resultar especialmente útil disponer de un buscador que permita no sólo realizar consultas por palabras clave, sino que sea capaz de filtrar los enlaces mostrados, permitiendo al usuario seleccionar una especialidad médica para restringir la búsqueda. Para ello nos bastaría clasificar previamente todas las páginas web que se indexaran, de forma que una vez categorizadas dentro del árbol MeSH, se le ofrezca al usuario la posibilidad de limitar los resultados a una rama específica. Respecto al motor de búsqueda, no resulta necesario que desarrollemos nuestra propia solución, ya que en la actualidad se dispone de aplicaciones de código abierto completamente operativas. Nos encontramos por ejemplo con YIOOP!,[2] la propuesta desarrollada por Chris Pollett [Patel et al., 2013], capaz de indexar cientos de millones de páginas web. De este modo proporcionaríamos una solución híbrida, ofreciendo las ventajas aportadas por la clasificación jerárquica de un directorio web y, a su vez, permitiendo recuperar o filtrar enlaces mediante palabras clave.

7.3. Publicaciones del autor

Parte del trabajo desarrollado en esta tesis ha sido publicado en diversos congresos. A continuación enumeramos las referencias bibliográficas, junto con una breve descripción del contenido de cada artículo. Hemos realizado una agrupación temática de cada una de las investigaciones para que sea más fácil identificar su propósito.

7.3.1. Algoritmo de clasificación de textos

En la primera de las siguientes publicaciones podemos encontrar una explicación de las bases conceptuales del algoritmo de clasificación desarrollado. Incluye resultados preliminares sobre las posibles mejoras que pueden proporcionar los planteamientos expuestos.

El segundo artículo describe ampliamente el algoritmo original de ranking de categorías. En este último caso proporcionamos comparaciones exhaustivas que demuestran la validez de los métodos utilizados.

- NEWPAR: an optimized feature selection and weighting schema for category ranking. *Frontiers in Artificial Intelligence and Applications, IOS Press, 2006* [Ruiz-Rico y Vicedo, 2006]. En este primer trabajo, aunque muy brevemente, ya se presentan algunas intuiciones para conseguir mejorar la selección y ponderación de características en el proceso de representación de documentos orientado a la obtención de un ranking de categorías.

[2] En http://www.yioop.com disponemos de una interfaz web para poder realizar búsquedas reales mediante YIOOP. Para obtener más detalles sobre el algoritmo utilizado, se puede consultar http://seekquarry.com, donde se dispone de la documentación completa del proyecto.

También se presentan algunas conclusiones positivas partiendo de evaluaciones puntuales sobre los corpus Reuters y OHSUMED.

- NEWPAR: an automatic feature selection and weighting schema for category ranking. *Proceedings of the 6th ACM symposium on Document Engineering, DocEng 2006* [Ruiz-Rico et al., 2006]. Proponemos observar en detalle colecciones de documentos típicas para analizar qué medidas y peculiaridades nos pueden resultar útiles para representar documentos y conseguir que las características resultantes sean lo más discriminativas y representativas posibles. Se presenta de forma detallada una simple combinación de diversas consideraciones, tales como el uso de sustantivos y adjetivos, n-gramas en vez de palabras, y la incorporación de medidas tales como la longitud de las expresiones. El resultado que se observa es un método sencillo de selección y ponderación de características que permite extraer, seleccionar y ponderar un tipo específico de n-gramas. Realizamos numerosos experimentos con diferentes conjuntos de documentos para comprobar la validez de la solución planteada.

7.3.2. Clasificación de artículos médicos

En las publicaciones que listamos a continuación ampliamos los detalles respecto a la evaluación del algoritmo de ponderación y selección de características aplicado y especializado en la clasificación de artículos médicos de la base de datos MEDLINE. En ellas podemos leer una versión extendida del algoritmo de suma de pesos evaluado mediante decisiones binarias y utilizando resúmenes de artículos médicos para los procesos de entrenamiento y evaluación.

Universitat d'Alacant
Universidad de Alicante

- Average-based discrimination for noun phrases selection and weighting applied to categorization of medline abstracts. *Proceedings of the 1st International Conference on Health Informatics, HEALTHINF 2008* [Ruiz-Rico et al., 2008b]. En este trabajo planteamos las ventajas y aplicaciones de la clasificación binaria, donde se decide si un determinado documento pertenece a cada una de las categorías de un corpus. En vez de evaluar el grado de pertenencia como hicimos previamente, ahora tratamos de tomar decisiones del tipo sí/no. Es el caso de la clasificación automática de artículos científicos dentro de la base de datos PubMed. Planteamos la posibilidad de entrenar el clasificador para poder decidir las categorías MeSH que deben ser asignadas a nuevas publicaciones. Observamos la validez de nuestra propuesta mediante diversas evaluaciones sobre OHSUMED. Vemos que el uso de nuestro sistema de representación de documentos utilizado anteriormente para ranking de categorías, también se puede utilizar de forma similar, construyendo un clasificador específico para cada categoría, de forma que podamos tomar decisiones de pertenencia. El uso del simple sistema de suma de pesos ya utilizado en los artículos publicados previamente, nos ayuda a plantear la mejora en la velocidad de respuesta respecto a algoritmos más complejos tales como SVM, mejorando incluso la calidad de las asignaciones finales.

- MEDLINE abstracts classification based on noun phrases extraction. *Biomedical Engineering Systems and Technologies. Communications in Computer and Information Science. Springer Berlin Heidelberg, 2009* [Ruiz-Rico et al., 2009]. El trabajo anterior fue considerado como uno de los mejores artículos publicados en la primera conferencia HEALTHINF. En consecuencia fue incluido en un libro que compendia diversos trabajos que fueron seleccionados.

7.3.3. Ayuda para el diagnóstico médico y prescripción de medicamentos y plantas medicinales

En la primera de las siguientes citas podemos encontrar el procedimiento para conseguir entrenar el algoritmo de clasificación de forma que se puedan procesar y obtener posibles diagnósticos médicos partiendo de historias clínicas escritas en lengua inglesa. En la segunda publicación que citamos a continuación, detallamos el proceso necesario para conseguir clasificar enfermedades y tratamientos mediante un entorno multilingüe, partiendo de síntomas en varios idiomas, y obteniendo la lista ordenada de categorías en el mismo idioma seleccionado para los datos de entrada. En el tercer artículo se obtiene una lista de plantas medicinales a partir de los síntomas.

- Medical diagnosis assistant based on category ranking. *Proceedings of the 1st International Conference on Health Informatics, HEALTHINF 2008* [Ruiz-Rico et al., 2008c]. El algoritmo de clasificación empieza a dar sus primeros frutos en el mundo real. Planteamos un sistema que permite la introducción de síntomas extraídos de historiales médicos para obtener como resultado un listado de posibles enfermedades ordenadas de mayor a menor probabilidad. Se trata de un clasificador de textos entrenado mediante cientos de miles de resúmenes de artículos médicos, cuya salida es un ranking de categorías, donde cada categoría es un término MeSH de la rama de enfermedades.

- Multilingual assistant for medical diagnosing and drug prescription based on category ranking. *Proceedings of the 22nd International Conference on Computational Linguistics, Coling 2008* [Ruiz-Rico et al., 2008a]. Realizamos diversas mejoras respecto a la primera propuesta del sistema de diagnóstico médico. Añadimos la posibilidad de introducir síntomas en diversos idiomas, ofreciendo a su vez un ranking de términos MeSH en el mismo idioma en el que se introdujeron los síntomas. Además, ofrecemos la posibilidad no sólo de obtener posibles enfermedades, sino que el usuario final puede optar por consultar también un listado de los principios activos de posibles medicamentos recomendados para los síntomas proporcionados.

- A multilingual and multiplatform application for medicinal plants prescription from medical symptoms. *Proceedings of the 36th ACM International Conference on Research and Development in Information Retrieval, SIGIR 2013* [Ruiz-Rico et al., 2013a]. En una tercera versión de la plataforma de clasificación de síntomas médicos, incluimos

la opción de poder obtener recomendaciones de plantas medicinales. Ampliamos también el abanico de dispositivos que nos permiten acceder a la aplicación, permitiendo no sólo su uso desde ordenadores portátiles o de sobremesa, sino desde muchas otras plataformas, tales como teléfonos móviles o tabletas.

7.3.4. Directorio médico utilizando artículos de la Wikipedia

En la siguiente serie de trabajos pretendemos aprovechar el potencial de nuestro algoritmo, que permite la clasificación de millones de documentos en tiempos muy reducidos.

- A portable multilingual medical directory by automatic categorization of Wikipedia articles. *Proceedings of the 36th ACM International Conference on Research and Development in Information Retrieval, SIGIR 2013* [Ruiz-Rico et al., 2013b]. En este trabajo destacamos el creciente uso de los dispositivos móviles y cómo esto nos empuja hacia el diseño de aplicaciones que permitan su ejecución mediante eventos táctiles, y preferiblemente sin la dependencia de una conexión a Internet. Proponemos la construcción de un directorio donde se pueda navegar a través de todo el árbol MeSH para alcanzar aquellos artículos de la Wikipedia que puedan estar relacionados con cada una de las categorías. Mediante nuestra solución permitimos obtener una aplicación completamente autónoma, que puede instalarse en una gran cantidad de sistemas operativos y dispositivos, y que muestra un breve resumen de cada artículo, requiriendo sólo conexión externa para consultar el contenido completo de los textos que se encuentran en la Wikipedia. Además, todo el proceso de construcción de la aplicación se realiza off-line, de forma que el acceso posterior es rápido y sin necesidad de conexión de datos.

Universitat d'Alacant
Universidad de Alicante

Universitat d'Alacant
Universidad de Alicante

Apéndice A

Descriptores de categorías

A.1. Reuters-21578, 90 categorías

- merger / acquisition
- aluminium
- barley
- bop / balance of payment
- carcass
- castor-oil / castor oil
- cocoa
- coconut
- coconut-oil / coconut oil
- coffee
- copper
- copra-cake / copra cake
- corn
- cotton
- cottonseed oil
- cpi / consumer price index
- cpu / capacity utilisation
- crude / crude oil
- dfl / netherlands guilder / florin
- dlr / dollar
- dmk / d-mark
- earn / earnings and earnings forecasts
- fuel / fuel oil
- gas / gasoline
- gnp / gross national / domestic product
- gold
- grain
- groundnut
- groundnut-oil / groundnut oil / groundnutseed oil
- heat / heating oil / gas oil
- hog
- housing / housing starts
- income / personal income

Universitat d'Alacant
Universidad de Alicante

- instal-debt / instalment debt / consumer credit
- interest / interest rate
- ipi / industrial production index
- iron-steel / iron / steel
- jet / kerosene
- job / unemployment
- l-cattle / live cattle
- lead
- lei / leading economic indicator
- lin-oil / linoil
- livestock
- lumber
- meal-feed / meal / feed
- money-fx / money / foreign exchange
- money-supply / money supply
- naphtha
- nat-gas / natural gas
- nickel
- nkr / norwegian krone / crown
- nzdlr / new zealand dollar
- oat
- oilseed
- orange
- palladium
- palmkernel
- palm-oil / palm oil
- palm-oil / petro-chemical
- platinum
- potato
- propane
- rand / south african rand
- rape-oil / rape oil
- rapeseed
- reserves
- retail / retail sale
- rice
- rubber
- rye
- ship / shipping
- silver
- sorghum
- soybean / soya bean
- soy-meal / soya meal / soymeal
- soy-oil / soya oil
- strategic-metal / strategic metal
- sugar
- sun-meal / sunflowerseed meal
- sun-oil / sunflowerseed oil
- sunseed
- tea
- tin
- trade
- unknown

Universitat d'Alacant
Universidad de Alicante

- veg-oil / vegetable oil
- wheat
- wpi / wholesale price index
- yen / japanese yen
- zinc

A.2. Reuters-21578, 50 categorías con mayor número de muestras

- merger/acquisition
- aluminium
- barley
- bop/balance of payment
- carcass
- cocoa
- coffee
- copper
- corn
- cotton
- cpi/consumer price index
- crude/crude oil
- dlr/dollar
- earn/earnings and earnings forecasts
- gas/gasoline
- gnp/gross national/domestic product
- gold
- grain
- housing/housing starts
- interest/interest rate
- ipi/industrial production index
- iron-steel/iron/steel
- job/unemployment
- livestock
- meal-feed/meal/feed
- money-fx/money/foreign exchange
- money-supply/money supply
- nat-gas/natural gas
- oilseed
- orange
- palm-oil/palm oil
- petro-chemical
- rapeseed
- reserves
- retail/retail sale
- rice
- rubber
- ship/shipping
- silver
- sorghum
- soybean/soya bean
- sugar

Universitat d'Alacant
Universidad de Alicante

- tin
- trade
- unknown
- veg-oil/vegetable oil
- wheat
- wpi/wholesale price index
- yen/japanese yen
- zinc

A.3. OHSUMED, sub-árbol de enfermedades

- Bacterial Infections and Mycoses / Bacterial Infections / ··· / Mucormycosis / Zoonoses
- Virus Diseases / Arbovirus Infections / ··· / Epidermodysplasia Verruciformis / Viremia/Zoonoses
- Parasitic Diseases / Central Nervous System Parasitic Infections / ··· / Lice Infestations / Scabies/Zoonoses
- Neoplasms / Cysts / Arachnoid Cysts / Bone Cysts / ··· / Sarcoma, Avian / Warts / Epidermodysplasia Verruciformis
- Musculoskeletal Diseases / Bone Diseases / Bone Cysts / ··· / Rheumatic Nodule / Wissler's Syndrome / Tennis Elbow
- Digestive System Diseases / Abdominal Pain / Abdomen, Acute / ··· / Pneumoperitoneum / Subphrenic Abscess / Tuberculosis, Peritoneal
- Stomatognathic Diseases / Jaw Diseases / Cherubism / ··· / Tooth Resorption / Root Resorption / Tooth, Unerupted / Toothache
- Respiratory Tract Diseases / Bronchial Diseases / Asthma / ··· / Tracheitis / Tracheobronchomegaly / Tracheoesophageal Fistula
- Otorhinolaryngologic Diseases / Ciliary Motility Disorders / ··· / Retropharyngeal Abscess / Tonsillitis / Velopharyngeal Insufficiency
- Nervous System Diseases / Autoimmune Diseases of the Nervous System / ··· / Spinal Cord Injuries / Autonomic Dysreflexia / Central Cord Syndrome
- Eye Diseases / Asthenopia / Conjunctival Diseases / Conjunctival Neoplasms / ··· / Vision, Low / Vitreoretinopathy, Proliferative / Vitreous Detachment
- Urologic and Male Genital Diseases / Fournier Gangrene / Genital Diseases, Male / ··· / Urinary Incontinence / Urinary Incontinence, Stress / Urinary Retention

Universitat d'Alacant
Universidad de Alicante

- Female Genital Diseases and Pregnancy Complications / Genital Diseases, Female / ··· / Galactorrhea / Mastitis / Postpartum Hemorrhage / Puerperal Infection
- Cardiovascular Diseases / Cardiovascular Abnormalities / Arterio-Arterial Fistula / ··· / Wegener's Granulomatosis / Venous Insufficiency / Postphlebitic Syndrome
- Hemic and Lymphatic Diseases / Hematologic Diseases / Anemia / ··· / Thymus Neoplasms / Tuberculosis, Lymph Node / King's Evil
- Congenital, Hereditary, and Neonatal Diseases and Abnormalities / Abnormalities / ··· / Thanatophoric Dysplasia / Toxoplasmosis, Congenital / Wolman Disease
- Skin and Connective Tissue Diseases / Connective Tissue Diseases / ··· / Miliaria / Fox-Fordyce Disease / Sweat Gland Neoplasms
- Nutritional and Metabolic Diseases / Metabolic Diseases / ··· / Starvation / Wasting Syndrome / HIV Wasting Syndrome
- Endocrine Diseases / Adrenal Gland Diseases / Adrenal Cortex Diseases / ··· / Thyrotoxicosis / Thyroid Crisis / Tuberculosis, Endocrine
- Immunologic Diseases / Autoimmune Diseases / Addison's Disease / ··· / Purpura, Thrombocytopenic / Purpura, Thrombocytopenic, Idiopathic
- Disorders of Environmental Origin / DNA Damage / Chromosome Breakage / ··· / Wounds, Gunshot / Wounds, Stab / Needlestick Injuries
- Animal Diseases / Abortion, Veterinary / Brucellosis, Bovine / ··· / Venereal Tumors, Veterinary / Wasting Disease, Chronic / Zoonoses
- Pathological Conditions, Signs and Symptoms / Pathological Conditions, Anatomical / ··· / Urological Manifestations / Anuria / Oliguria / Polyuria

A.4. OHSUMED, sub-árbol de enfermedades del corazón

- Heart Diseases
- Arrhythmia
- Arrhythmia, Sinus
- Atrial Fibrillation
- Atrial Flutter
- Bradycardia
- Heart Block
- Adams-Stokes Syndrome
- Bundle-Branch Block
- Sinoatrial Block
- Long QT Syndrome
- Pre-Excitation Syndromes

Universidad de Alicante

- Pre-Excitation, Mahaim-Type
- Wolff-Parkinson-White Syndrome
- Sick Sinus Syndrome
- Tachycardia
- Tachycardia, Paroxysmal
- Tachycardia, Supraventricular
- Tachycardia, Atrioventricular Nodal Reentry
- Tachycardia, Ectopic Atrial
- Tachycardia, Ectopic Junctional
- Tachycardia, Sinoatrial Nodal Reentry
- Tachycardia, Sinus
- Ventricular Fibrillation
- Carcinoid Heart Disease
- Cardiac Output, Low
- Cardiac Tamponade
- Cardiomyopathy, Congestive
- Coronary Disease
- Angina Pectoris
- Angina, Unstable
- Angina Pectoris, Variant
- Coronary Aneurysm
- Coronary Arteriosclerosis
- Coronary Thrombosis
- Coronary Vasospasm
- Endocarditis
- Endocarditis, Bacterial
- Endocarditis, Subacute Bacterial
- Heart Aneurysm
- Heart Arrest
- Heart Defects, Congenital
- Aortic Coarctation
- Coronary Vessel Anomalies
- Cor Triatriatum
- Crisscross Heart
- Dextrocardia
- Ductus Arteriosus, Patent
- Ebstein's Anomaly
- Eisenmenger Complex
- Heart Septal Defects
- Aortopulmonary Septal Defect
- Endocardial Cushion Defects
- Heart Septal Defects, Atrial
- Heart Septal Defects, Ventricular
- Levocardia
- Marfan Syndrome
- Tetralogy of Fallot
- Transposition of Great Vessels
- Double Outlet Right Ventricle
- Truncus Arteriosus, Persistent
- Heart Failure, Congestive
- Dyspnea, Paroxysmal
- Edema, Cardiac
- Heart Neoplasms

Universitat d'Alacant
Universidad de Alicante

- Heart Rupture
- Heart Rupture, Post-Infarction
- Heart Valve Diseases
- Aortic Valve Insufficiency
- Aortic Valve Prolapse
- Aortic Valve Stenosis
- Cardiomyopathy, Hypertrophic
- Heart Murmurs
- Mitral Valve Insufficiency
- Mitral Valve Prolapse
- Mitral Valve Stenosis
- Pulmonary Valve Insufficiency
- Pulmonary Valve Stenosis
- Pulmonary Subvalvular Stenosis
- Tricuspid Valve Insufficiency
- Tricuspid Valve Prolapse
- Tricuspid Valve Stenosis
- Myocardial Diseases
- Cardiomyopathy, Alcoholic
- Cardiomyopathy, Restrictive
- Chagas Cardiomyopathy
- Endocardial Fibroelastosis
- Endomyocardial Fibrosis
- Kearns Syndrome
- Myocardial Reperfusion Injury
- Myocarditis
- Myocardial Infarction
- Shock, Cardiogenic
- Pericardial Effusion
- Pericarditis
- Pericarditis, Constrictive
- Pericarditis, Tuberculous
- Pneumopericardium
- Postpericardiotomy Syndrome
- Pulmonary Heart Disease
- Rheumatic Heart Disease
- Ventricular Outflow Obstruction

A.5. OHSUMED, 50 categorías con mayor número de muestras

- Adult
- Middle Age
- Aged
- Child
- Rats
- Child, Preschool
- Mice
- Pregnancy

Universitat d'Alacant
Universidad de Alicante

- Follow-Up Studies
- Infant
- Aged, 80 and over
- Rats, Inbred Strains
- Infant, Newborn
- Kinetics
- Clinical Trials
- Prospective Studies
- Postoperative Complications
- Risk
- Dogs
- Random Allocation
- Blood Pressure
- Prognosis
- Antibodies, Monoclonal
- Cell Line
- Age Factors
- Liver
- Cells, Cultured
- Retrospective Studies
- Time Factors
- Diagnosis, Differential
- T-Lymphocytes
- Hemodynamics
- Brain
- DNA
- Kidney
- Lung
- Dose-Response Relationship, Drug
- Rabbits
- Calcium
- Evaluation Studies
- Skin
- Hypertension
- Muscles
- Microscopy, Electron
- Base Sequence
- Combined Modality Therapy
- Acute Disease
- Antigens, Surface
- Tomography, X-Ray Computed
- Electrocardiography

Universitat d'Alacant
Universidad de Alicante

Apéndice B

Palabras de parada

A continuación enumeramos las palabras de parada seleccionadas automáticamente después del proceso de aprendizaje realizado para cada uno de los corpus. Entre paréntesis indicamos la cantidad promediada de apariciones (TF) y de documentos (DF) en los que aparece dicho término.[1] Por limitaciones de espacio, mostraremos un máximo de 100 términos en cada lista, ordenándolos de mayor a menor cantidad de apariciones.

B.1. Reuters-21578, 90 categorías

to (731,124)	have (213,79)	by (140,64)	but (83,46)
of (609,117)	it (205,90)	trade (136,45)	are (81,45)
in (537,108)	u (187,66)	with (135,68)	sayyid the (80,53)
and (445,104)	s (185,62)	new (127,72)	april (79,57)
be (434,95)	year (179,83)	market (120,51)	last (75,43)
sayyid (410,108)	on (174,72)	would (115,49)	also (65,45)
a (393,95)	march (169,132)	not (112,59)	expect (64,39)
mln (363,103)	that (167,60)	price (106,41)	report (64,33)
the (293,102)	its (153,76)	an (100,60)	up (62,41)
dlrs (267,95)	at (144,69)	as (98,49)	more (58,37)
for (254,94)	from (144,65)	this (97,53)	month (52,34)
pct (237,70)	will (142,65)	which (88,55)	than (52,34)

B.2. Reuters-21578, 50 categorías con mayor número de muestras

[1] Estos valores promediados de TF y DF se calculan sumando las repeticiones totales de cada término para todos los documentos que pertenecen a cada categoría, y dividiendo dicho valor entre el número total de categorías en las que aparece dicho término.

mln (460,131)
dlrs (320,115)
pct (308,92)
billion (282,86)
u (250,88)
s (240,81)
march (229,179)
year (214,100)
at (206,99)
oil (198,55)
rate (183,61)
tonne (177,43)
trade (161,52)
market (148,62)
sale (127,60)
price (125,46)
japan (115,40)
export (102,42)
are (100,55)
new (96,73)
april (96,70)
february (87,31)
may (85,53)
week (80,41)
up (77,51)
buy (73,51)
plan (72,44)
government (72,37)
import (71,28)
january (69,27)
country (66,35)
today (64,49)
world (64,32)
official (64,31)
month (63,41)
other (61,43)
production (61,26)
see (58,49)
state (57,34)
report (57,30)
last year (54,34)
washington (52,46)
over (52,38)
rise (52,32)
increase (51,34)
down (46,31)
department (43,26)
end (42,34)
fall (42,28)
reuters (41,32)
now (40,31)
level (39,27)
while (37,30)
demand (34,21)

B.3. OHSUMED, sub-árbol de enfermedades

patients (1255,297)
cell (344,93)
disease (339,141)
at (301,169)
study (276,175)
group (272,98)
treatment (256,125)
case (248,133)
are (225,153)
p (217,84)
tumour (206,63)
year (191,100)
effect (187,104)
infection (187,70)
cancer (175,56)
blood (174,71)
carcinoma (172,62)
result (166,125)
may (165,121)
syndrome (159,67)
therapy (152,76)
artery (151,57)
level (147,69)
patient (141,96)
heart (139,53)
rat (138,37)
pressure (137,48)
child (137,43)
rate (136,72)
control (136,70)
injury (134,45)
lung (133,46)
pain (132,44)
risk (130,59)
factor (123,66)
month (117,65)
woman (117,46)
age (110,67)
surgery (109,54)
other (108,88)
survival (108,45)
virus (108,44)
complication (106,58)
lesion (106,48)
day (105,52)
serum (105,43)
diagnosis (104,70)
response (104,52)
flow (100,36)
glucose (99,22)
change (97,61)
death (97,41)
coronary artery (96,40)
subject (96,38)
hypertension (96,37)
time (94,62)
type (94,49)
antibody (93,33)
test (92,49)
b (92,32)
failure (91,45)
report (89,73)
use (89,64)
activity (89,42)
breast (88,23)
function (87,50)
analysis (85,60)
concentration (85,38)
n (85,30)
toilet (84,45)
ischaemia (83,31)
can (82,65)
liver (82,30)
mg (81,36)
plasma (81,30)

Universitat d'Alacant
Universidad de Alicante

diabetes (79,29)
infarction (79,29)
receptor (79,25)
mortality (78,40)
pregnancy (78,26)
mean (77,56)
immunodeficiency (76,35)
image (75,27)
graft (75,23)
finding (74,57)
increase (74,49)
aid (74,25)
catheter (73,22)
mouse (73,18)
i (72,36)
bone (72,27)
symptom (71,44)
protein (71,27)
muscle (71,26)
gene (71,23)
infant (71,23)
valve (71,22)
bladder (71,21)
blood pressure (70,27)
dna (70,22)
...

B.4. OHSUMED, sub-árbol de enfermedades del corazón

artery (371,133)
heart (297,110)
disease (268,109)
group (261,73)
p (248,85)
at (247,118)
valve (240,62)
effect (211,104)
study (162,100)
rate (150,72)
year (133,63)
death (126,58)
patients (124,165)
are (114,76)
treatment (112,62)
result (100,73)
mean (97,63)
time (95,57)
may (88,64)
month (87,51)
case (78,47)
patient (74,52)
other (67,54)
finding (51,40)
four (51,38)

B.5. OHSUMED, 50 categorías con mayor número de muestras

Universitat d'Alacant
Universidad de Alicante

at (1078,574)
study (905,575)
rat (873,243)
effect (860,435)
group (825,308)
patients (796,846)
disease (791,343)
treatment (720,361)
p (654,254)
are (646,433)
level (573,254)
year (572,295)
t (557,117)
case (542,289)
blood (511,213)
result (499,377)
pressure (498,169)
response (494,220)
tumour (491,150)
rate (485,253)
child (484,161)
mouse (468,136)
antibody (463,146)
age (438,224)
cell (437,346)
may (419,330)
therapy (419,203)
activity (419,169)
factor (415,205)
control (414,227)
protein (405,131)
cancer (399,132)
receptor (382,109)
risk (375,166)
change (370,218)
artery (359,135)
plasma (357,125)
day (354,176)
infection (344,133)
concentration (343,157)
serum (340,137)
other (337,269)
b (331,100)
gene (331,94)
month (327,184)
mg (324,147)
t cell (307,83)
antigen (304,104)
increase (302,192)
heart (302,129)
time (300,205)

carcinoma (299,110)
diagnosis (297,184)
woman (294,126)
dose (287,134)
test (285,136)
ct (285,83)
function (281,165)
patient (280,201)
exercise (280,66)
syndrome (278,122)
insulin (274,48)
surgery (272,148)
complication (271,154)
c (270,95)
liver (267,90)
infant (267,81)
valve (262,58)
type (261,132)
lung (261,85)
subject (257,122)
hypertension (257,101)
analysis (255,186)
muscle (255,79)
pain (251,92)
follow-up (250,170)
toilet (249,129)
i (248,116)
injury (247,83)
mean (246,172)
survival (245,103)
mortality (244,111)
bone (241,80)
flow (240,80)
drug (237,116)
fracture (237,60)
h (235,90)
image (233,74)
finding (229,181)
growth (228,88)
calcium (227,65)
infarction (225,83)
value (224,146)
difference (223,166)
four (222,166)
use (222,160)
lesion (221,105)
blood pressure (220,90)
pregnancy (219,76)
expression (218,92)
...

B.6. Diagnóstico médico

lymphoma (468,146)
infection (331,158)
carcinoma (322,264)
cancer (319,137)
liver (305,112)
case (261,24)
year (259,184)
tumour (255,318)
case report (251,234)
syndrome (246,229)
cell (244,266)
bone (234,113)
literature (233,178)
pregnancy (233,85)
pain (232,115)
study (223,166)
complication (216,147)
finding (210,163)
child (205,107)
revue (193,156)
image (192,105)
are (192,62)
muscle (191,87)
patient (188,184)
mass (179,110)
lung (179,88)
brain (178,93)
other (177,150)
woman (176,145)
surgery (176,118)
report (176,17)
failure (175,101)
biopsy (173,117)
month (171,129)
p (170,59)
blood (169,109)
feature (168,128)
type (168,102)
chemotherapy (166,98)
antibody (163,80)
toilet (162,105)
level (157,95)
transplantation (157,69)
t (157,68)
result (153,130)
cause (153,126)
symptom (153,115)
skin (149,85)
c (149,62)
resection (147,95)
factor (146,85)
man (142,124)
can (142,118)
ct (142,78)
b (140,67)
treatment (137,252)
management (137,101)
neoplasm (136,91)
gene (134,58)
disorder (132,92)
day (131,88)
serum (131,76)
examination (130,108)
involvement (129,90)
malignancy (127,84)
use (119,90)

Universitat d'Alacant
Universidad de Alicante

infant (118,66)
four (117,85)
plasma (117,58)
tomography (112,85)
change (112,81)
drug (110,73)
family (110,57)
manifestation (109,83)
association (108,85)
abnormality (108,79)
system (108,74)
effect (107,78)
analysis (105,74)
week (103,78)
resonance (102,74)
procedure (101,75)
magnetic resonance (101,73)
diagnosis (100,248)
evidence (100,88)
condition (100,86)
technique (100,72)
presentation (99,82)
response (98,67)
course (97,83)
history (96,84)
scan (95,67)
recurrence (94,68)
protein (93,50)
time (92,77)
function (92,69)
method (91,79)
area (91,70)
risk (91,68)
test (91,63)
...

Universitat d'Alacant
Universidad de Alicante

Universitat d'Alacant
Universidad de Alicante

Apéndice C

Influencia de los parámetros en el ranking de categorías

En las tablas C.1, C.2, C.3, C.4, C.5, C.6, podemos observar los resultados detallados y el deterioro resultante al prescindir de cada uno de los parámetros utilizando como algoritmo de clasificación la suma de pesos. La evaluación se ha llevado a cabo utilizando las medidas OneErr, IsErr, AvgP, BEP, maxF1 y Cov.

Universitat d'Alacant
Universidad de Alicante

Parámetro	OneErr	IsErr	AvgP	BEP	maxF1	Cov
-	8.09	11.85	94.29	90.69	95.56	0.81
CF	15.10	19.49	89.50	83.61	91.80	1.25
Frases	11.43	16.10	91.72	87.05	93.42	0.96
Frases.PoS	10.76	15.37	92.13	87.79	93.76	1.12
TFnew	11.82	16.10	91.92	87.08	93.76	0.92
Des.Cat.	10.40	14.25	92.57	88.45	94.20	0.97
TL	10.43	14.19	92.75	88.58	94.39	0.89
Diccionario	10.00	13.97	92.95	88.86	94.48	0.90
Títulos	10.31	14.28	93.04	88.60	94.64	0.80
TF	8.79	12.73	93.85	90.08	95.23	0.84
Tagger	8.61	12.37	94.07	90.33	95.42	0.82
DF	8.46	12.12	94.11	90.42	95.43	0.81

Tabla C.1: Resultados para Reuters después de aislar y quitar la influencia de cada uno de los parámetros del algoritmo.

Universitat d'Alacant

Parámetro	OneErr	IsErr	AvgP	BEP	maxF1	Cov	**Media**
CF	86.52	64.45	5.08	7.80	3.94	55.06	**37.1**
Frases	41.20	35.81	2.73	4.01	2.23	19.07	**17.5**
Frases.PoS	32.96	29.67	2.30	3.20	1.88	38.32	**18.1**
TFnew	46.07	35.81	2.51	3.98	1.88	13.75	**17.3**
Des.Cat.	28.46	20.20	1.82	2.47	1.42	20.34	**12.5**
TL	28.84	19.69	1.64	2.32	1.22	10.15	**10.6**
Diccionario	23.59	17.90	1.42	2.01	1.12	11.27	**9.6**
Títulos	27.34	20.46	1.32	2.30	0.96	-1.08	**8.6**
TF	8.61	7.42	0.46	0.67	0.34	3.71	**3.5**
Tagger	6.36	4.34	0.23	0.39	0.14	0.82	**2.1**
DF	4.49	2.30	0.19	0.30	0.13	-0.41	**1.2**

Tabla C.2: Deterioro en el funcionamiento (%) para Reuters después de aislar y quitar la influencia de cada uno de los parámetros del algoritmo.

Parámetro	OneErr	IsErr	AvgP	BEP	maxF1	Cov
-	21.49	34.94	82.58	73.72	85.71	2.11
CF	29.70	43.33	76.98	66.14	81.46	2.73
Frases	30.87	45.43	75.79	64.17	80.32	2.85
Frases.PoS	27.28	40.97	78.62	68.20	82.60	2.63
TFnew	27.47	39.69	79.43	69.13	83.58	2.25
Des.Cat.	23.46	36.82	81.35	72.02	84.82	2.22
TL	24.42	38.20	80.57	70.90	84.17	2.27
Diccionario	22.40	35.79	81.92	72.96	85.21	2.25
Títulos	23.78	36.53	81.58	72.22	85.08	2.18
TF	21.79	35.10	82.38	73.58	85.59	2.13
Tagger	21.89	35.23	82.30	73.35	85.47	2.16
DF	21.63	35.31	82.44	73.34	85.60	2.12

Tabla C.3: Resultados para OHSUMED (enfermedades del corazón) después de aislar y quitar la influencia de cada uno de los parámetros del algoritmo.

Universitat d'Alacant

Parámetro	OneErr	IsErr	AvgP	BEP	maxF1	Cov	**Media**
CF	38.20	24.03	6.78	10.27	4.95	29.69	**19.0**
Frases	43.63	30.04	8.23	12.94	6.29	35.24	**22.7**
Frases.PoS	26.95	17.26	4.80	7.49	3.62	25.10	**14.2**
TFnew	27.81	13.61	3.81	6.22	2.48	7.00	**10.2**
Des.Cat.	9.15	5.40	1.49	2.30	1.04	5.51	**4.2**
TL	13.60	9.35	2.43	3.82	1.80	7.71	**6.5**
Diccionario	4.20	2.43	0.80	1.03	0.58	6.64	**2.6**
Títulos	10.63	4.56	1.21	2.02	0.73	0.58	**3.3**
TF	1.36	0.46	0.24	0.19	0.14	1.01	**0.6**
Tagger	1.85	0.84	0.34	0.49	0.28	2.79	**1.10**
DF	0.62	1.06	0.17	0.51	0.13	0.60	**0.5**

Tabla C.4: Deterioro en el funcionamiento (%) para OHSUMED (enfermedades del corazón) después de aislar y quitar la influencia de cada uno de los parámetros del algoritmo.

Parámetro	OneErr	IsErr	AvgP	BEP	maxF1	Cov
-	22.05	43.46	81.35	70.95	84.91	2.15
CF	25.53	47.40	78.67	67.38	82.90	2.37
Frases	27.78	50.44	76.60	64.78	81.05	2.61
Frases.PoS	25.08	46.09	79.67	68.54	83.32	2.34
TFnew	24.18	45.05	80.26	69.26	84.30	2.19
Des.Cat.	24.71	46.28	79.44	68.34	83.53	2.30
TL	23.62	44.83	80.29	69.53	84.11	2.22
Diccionario	23.06	44.52	80.50	69.85	84.04	2.26
Títulos	22.15	42.90	81.66	71.12	84.85	2.12
TF	22.28	43.56	81.32	70.83	84.89	2.15
Tagger	22.15	43.29	81.38	71.00	84.55	2.15
DF	22.31	43.92	81.08	70.57	84.64	2.19

Tabla C.5: Resultados para OHSUMED (enfermedades) después de aislar y quitar la influencia de cada uno de los parámetros del algoritmo.

Universitat d'Alacant
Universidad de Alicante

Parámetro	OneErr	IsErr	AvgP	BEP	maxF1	Cov	**Media**
CF	15.79	9.06	3.29	5.11	2.36	10.22	**7.6**
Frases	26.01	16.06	5.84	8.69	4.54	20.97	**13.7**
Frases.PoS	13.75	6.07	2.06	3.40	1.87	8.39	**5.9**
TFnew	9.67	3.67	1.34	2.38	0.72	1.76	**3.3**
Des.Cat.	12.05	6.50	2.35	3.68	1.62	6.63	**5.5**
TL	7.12	3.16	1.31	2.00	0.95	3.20	**3.0**
Diccionario	4.57	2.44	1.05	1.55	1.02	4.72	**2.6**
Títulos	0.46	-1.28	-0.38	-0.24	0.07	-1.63	**-0.5**
TF	1.07	0.24	0.04	0.16	0.02	-0.42	**0.2**
Tagger	0.46	-0.37	-0.03	-0.08	0.43	0.00	**0.1**
DF	1.19	1.05	0.34	0.54	0.32	1.47	**0.8**

Tabla C.6: Deterioro en el funcionamiento (%) para OHSUMED (enfermedades) después de aislar y quitar la influencia de cada uno de los parámetros del algoritmo.

Apéndice D

Peso óptimo para las expresiones de los títulos

En las tablas D.1, D.2, D.3, D.4, D.5 y D.6, observamos los resultados detallados y el deterioro producido al aplicar diferentes factores de ponderación para las expresiones de los títulos. Los valores óptimos se han resaltado en negrita.

Universitat d'Alacant
Universidad de Alicante

Factor	OneErr	IsErr	AvgP	BEP	maxF1	Cov
1	10.31	14.28	93.04	88.60	94.64	**0.80**
2	**8.09**	**11.85**	**94.29**	**90.69**	**95.56**	0.81
3	9.12	12.79	93.63	89.79	95.06	0.85
4	9.40	13.06	93.41	89.53	94.89	0.90
5	9.91	13.58	92.94	88.98	94.52	0.99

Tabla D.1: Evaluación de los resultados utilizando Reuters y considerando diferentes factores de ponderación para las expresiones de los títulos.

Factor	OneErr	IsErr	AvgP	BEP	maxF1	Cov	**Media**
1	27.34	20.46	1.32	2.30	0.96	-1.08	**8.6**
3	12.73	7.93	0.70	0.99	0.52	5.29	**4.7**
4	16.10	10.23	0.93	1.28	0.70	11.39	**6.8**
5	22.47	14.58	1.43	1.88	1.08	21.88	**10.6**

Tabla D.2: Deterioro en el funcionamiento (%) para Reuters aplicando diferentes factores de ponderación para las expresiones de los títulos.

Factor	OneErr	IsErr	AvgP	BEP	maxF1	Cov
1	23.78	36.53	81.58	72.22	85.08	2.12
2	21.49	**34.94**	**82.58**	**73.72**	85.71	**2.11**
3	**21.44**	35.07	82.56	73.44	**85.72**	2.12
4	21.76	35.81	82.23	72.95	85.40	2.18
5	21.76	36.00	82.11	72.92	85.26	2.22

Tabla D.3: Evaluación de los resultados utilizando OHSUMED (enfermedades del corazón) y considerando diferentes factores de ponderación para las expresiones en los títulos.

Factor	OneErr	IsErr	AvgP	BEP	maxF1	Cov	**Media**
1	10.63	4.56	1.21	2.02	0.73	0.58	**3.3**
3	-0.25	0.38	0.02	0.38	-0.01	0.62	**0.2**
4	1.24	2.51	0.43	1.04	0.36	3.73	**1.6**
5	1.24	3.04	0.57	1.07	0.53	5.44	**2.0**

Tabla D.4: Deterioro en el rendimiento (%) para OHSUMED (enfermedades del corazón) aplicado diferentes factores de ponderación para las expresiones en los títulos.

Factor	OneErr	IsErr	AvgP	BEP	maxF1	Cov
1	22.15	**42.90**	**81.66**	**71.12**	84.85	**2.12**
2	**22.05**	43.46	81.35	70.95	**84.91**	2.15
3	22.93	44.35	80.54	70.03	84.32	2.24
4	23.05	45.15	80.05	69.48	83.43	2.30
5	24.07	46.00	79.36	68.60	82.95	2.35

Tabla D.5: Resultados para OHSUMED (enfermedades) considerando diferentes factores de ponderación para las expresiones en los títulos.

Factor	OneE.	IsErr	AvgP	BEP	maxF1	Cov	**Media**
1	0.46	-1.28	-0.38	-0.24	0.07	-1.63	**-0.5**
3	3.98	2.05	1.00	1.29	0.69	3.91	**2.2**
4	4.56	3.90	1.59	2.07	1.74	6.74	**3.4**
5	9.18	5.86	2.44	3.31	2.31	9.26	**5.4**

Tabla D.6: Deterioro en el funcionamiento (%) para OHSUMED (enfermedades) aplicando diferentes factores de ponderación para las expresiones en los títulos.

Apéndice E

Ejemplos adicionales de diagnóstico médico

A continuación procedemos a detallar dos ejemplos adicionales que reflejan el funcionamiento del algoritmo de clasificación para proporcionar diagnósticos médicos de forma automática partiendo de historias clínicas reales.

E.1. Ejemplo 3: Embolia pulmonar

El diagnóstico proporcionado por los expertos para la historia clínica de la figura E.1 es "Embolia pulmonar". Transladando esto a una o varias de las 23 enfermedades del nivel más alto del MeSH, se obtienen las siguientes entradas:

- "Enfermedades Cardiovasculares"
- "Enfermedades Respiratorias"

La figura E.2 muestra la salida del sistema para la anterior historia clínica, obteniendo los siguientes valores tras su evaluación:

- $P_{r=0} = 0$, puesto que el primer elemento no coincide con ninguo de los proporcionados por los expertos
- $AvgP = \frac{\frac{1}{2} + \frac{2}{3}}{2} = 0,58$, puesto que los diagnósticos correctos se hayan en las posiciones 2 y 3
- $BEP = \frac{2}{3} = 0,67$

Hombre de 75 años, con antecedentes de enfisema pulmonar, que acudió al hospital tras sufrir síncope, presentando disnea de reposo e hipotensión al recuperar conciencia.

A su llegada a urgencias se encontraba consciente, sudoroso, taquipneico, con hipotensión que requirió la administración de dopamina a dosis beta, hipoxemia e hipercapnia. En el ECG se observó ritmo sinusal con sobrecarga de cavidades derechas y bloqueo incompleto de rama derecha, que no existía en ECG previos. Se realizó TC torácica helicoidal, en la que se visualizaron imágenes de trombo en ambos troncos pulmonares y arterias lobares bilaterales, así como una imagen "en silla de montar" entre ambos troncos arteriales pulmonares, sobre un parénquima pulmonar muy dañado, con enfisema bulloso y fibrosis pulmonar importantes.

Se administró un bolo de 10 mg de rt-PA seguido de 90 mg en perfusión continua durante dos horas, presentando (a falta de 10 mg) hemoptisis y hematomas en miembros superiores, por lo que se decidió interrumpir en ese momento la fibrinolisis y empezar tratamiento con heparina a dosis anticoagulantes. Pasadas 48 horas el enfermo presentó empeoramiento de la insuficiencia respiratoria; se realizó ecocardiograma, que mostró un ventrículo derecho hipertrófico, dilatado e hipocontráctil con una PSAP de 55 mmHg, planteándose la necesidad de fibrinolisis de rescate o trombectomía endovascular, y planificándose para el día siguiente este último procedimiento. Previamente a dicha intervención se realizó nueva TC helicoidal, que mostró una evolución favorable con respecto al previo, con imágenes de trombo residual en arterias segmentarias pero no en arterias principales o lobares, por lo que se suspendió la trombectomía endovascular y se continuó con anticoagulación al atribuir el cuadro clínico al importante enfisema que presentaba el paciente. Tras doce días de ingreso en UCI el enfermo fue trasladado a planta y poco después fue dado de alta hospitalaria.

Figura E.1: Historial médico de paciente con embolia pulmonar. El texto completo del caso se puede obtener de http://remi.uninet.edu/casos/caso5/REMIC005.htm.

Síntomas:

Hombre de 75 años, con antecedentes de enfisema pulmonar, que acudió al hospital tras sufrir síncope, presentando disnea de reposo e

Comprobar | Plantas medicinales

○ Rápido ◉ Por jerarquías

Más información:

administración hospitalaria; insuficiencia respiratoria; enfisema pulmonar; fibrosis pulmonar; anticoagulantes; bloqueo de rama; fibrinólisis; trombectomía; hipercapnia; hipotensión; ritmo beta; conciencia; hemoptisis; evolución; evolución; perfusión; arterias; dopamina; enfisema; fibrosis; heparina; síncope; disnea

Orden	Enfermedades	Valoración	Bibliogr.	Expandir
1	Condiciones Patológicas, Signos y Síntomas	100	+ +	->
2	Enfermedades Respiratorias	71	+ +	->
3	Enfermedades Cardiovasculares	58	+ +	->
4	Enfermedades del Sistema Nervioso	30	+ +	->
5	Enfermedades Neonatales Congénitas y Hereditarias y Anomalías	14	+ +	->
6	Enfermedades de la Piel y Tejido Conjuntivo	10	+ +	->
7	Enfermedades Hematológicas y Linfáticas	10	+ +	->
8	Trastornos de Origen Ambiental	8	+ +	->
9	Neoplasias	6	+ +	->
10	Enfermedades Nutricionales y Metabólicas	6	+ +	->
11	Enfermedades Urogenitales Femeninas y Complicaciones del Embarazo	6	+ +	->
12	Enfermedades del Sistema Digestivo	5	+ +	->
13	Enfermedades del Sistema Inmune	5	+ +	->
14	Enfermedades Musculoesqueléticas	4	+ +	->
15	Oftalmopatías	3	+ +	->
16	Enfermedades del Sistema Endocrino	3	+ +	->
17	Enfermedades Urogenitales Masculinas	3	+ +	->
18	Infecciones Bacterianas y Micosis	3	+ +	->
19	Trastornos Mentales	2	+ +	->
20	Enfermedades Estomatognáticas	1	+ +	->
21	Enfermedades Otorrinolaringológicas	1	+ +	->

Universitat d'Alacant
Universidad de Alicante

Figura E.2: Ejemplo del primer nivel de diagnóstico jerárquico para un caso de embolia pulmonar (correspondiente al historial de la figura E.1, página 192).

E.2. Ejemplo 4: Ataxia

El diagnóstico proporcionado por los expertos para la historia clínica de la figura E.3 es "Ataxia". Transladando esto a una o varias de las 23 enfermedades del nivel más alto del MeSH, se obtienen las siguientes entradas:

- "Enfermedades del Sistema Nervioso"
- "Condiciones Patológicas, Signos y Síntomas"

La figura E.4 muestra la salida del sistema para la anterior historia clínica, obteniendo los siguientes valores tras su evaluación:

- $P_{r=0} = 1$, puesto que el primer elemento ("Enfermedades del Sistema Nervioso" coincide con uno de los diagnósticos dados por los expertos
- $AvgP = \frac{\frac{1}{1} + \frac{2}{2}}{2} = 1$, puesto que los diagnósticos correctos se hayan en las posiciones 1 y 2
- $BEP = \frac{2}{2} = 1$

Universitat d'Alacant
Universidad de Alicante

Paciente de 34 años que es remitido para estudio por un trastorno de la marcha.

- *Antecedentes personales:*
  - *Historia familiar negativa.*
  - *Fumador de un paquete día.*
  - *No bebedor.*
  - *Hizo el servicio militar sin problemas.*
  - *Albañil activo hasta la actualidad (pero es trabajador por cuenta propia).*
- *Enfermedad actual: Sin que pueda precisar una fecha concreta, pues el cuadro se ha caracterizado a lo largo de todo su curso por una evolución lenta, insidiosa, parece ser que hace unos 5 años comenzó a notar una inestabilidad que no le permitía manejar la bicicleta con seguridad. Comenzó con una marcha torpe y hace tres años tuvo que dejar el footing por estos problemas. Se siguó cierta torpeza en la manipulación de objetos, más tarde una sutil alteración del tono de voz que sólo notaban los amigos que hacía tiempo que no le veían. Ultimamente los movimientos más precisos estaban muy afectados, casi imposible la escritura. Ha dejado ahora su actividad laboral. No diplopia, disfagia, alteraciones sensitivas, cognitivas o esfinterianas.*
- *Exploración:*
  - *Estado mental : normal.*
  - *Lenguage: Disartria cerebelosa que no impide comprenderle con facilidad.*
  - *Motor: tono y fuerza normales.*
  - *Sensibilidad: Táctil y posicional conservadas.*
  - *Reflejos: Bicipital , estilorradial, rotulianos + (hipoactivos), aquíleos ++ (normales).*
  - *Cerebelo: Dismetría moderada en extremidades. Marcha con aumento de la base de sustentación, torpe, tandem imposible.*
- *Pruebas complementarias:*
  - *RMN cerebral: normal (quizá puente-bulbo ligeramente disminuidos de tamaño).*
  - *RNM medular: atrofia del cordón medular.*
  - *EMG-ENG: Descenso moderado-severo de la amplitud del componente sensitivo evocado del N. Mediano Izquierdo (III dedo), del N. Cubital izquierdo (V dedo), N. Radial (I dedo). N. Peroneal derecho (componente motor): normal. M. Pedio derecho: no denervación.*
- *Analítica:*
  - *(Valores normales o negativos): RPR, TSH, hemograma, bioquímica de rutina, ANA, C3 y C4, FR, lactato.*
  - *Pendiente de niveles de vitamina E.*
  - *Se ha remitido muestra para estudio genético.*

Figura E.3: Historial médico de paciente con ataxia. El texto completo del caso se puede obtener de http://neurologia.rediris.es/neurologia/casos/caso14.html.

Síntomas:

Paciente de 34 años que es remitido para estudio por un trastorno de la marcha. antecedentes personales historia familiar negativa. fumador de

Comprobar | Plantas medicinales

○ Rápido ◉ Por jerarquías

Más información:

tamaño de la muestra; extremidades; bioquímica; enfermedad; vitamina a; vitamina e; disartria; evolución; evolución; seguridad; escritura; cerebelo; diplopía; historia; atrofia; amigos; marcha; rutina; tiempo; voz

Orden	Enfermedades	Valoración	Bibliogr.	Expandir
1	Enfermedades del Sistema Nervioso	100	+ +	->
2	Condiciones Patológicas, Signos y Síntomas	80	+ +	->
3	Enfermedades Nutricionales y Metabólicas	58	+ +	->
4	Enfermedades Neonatales Congénitas y Hereditarias y Anomalías	27	+ +	->
5	Oftalmopatías	21	+ +	->
6	Neoplasias	20	+ +	->
7	Enfermedades Respiratorias	16	+ +	->
8	Enfermedades Hematológicas y Linfáticas	16	+ +	->
9	Enfermedades del Sistema Digestivo	14	+ +	->
10	Enfermedades Cardiovasculares	13	+ +	->
11	Enfermedades Musculoesqueléticas	13	+ +	->
12	Enfermedades Otorrinolaringológicas	12	+ +	->
13	Trastornos Mentales	12	+ +	->
14	Enfermedades de la Piel y Tejido Conjuntivo	12	+ +	->
15	Trastornos de Origen Ambiental	10	+ +	->
16	Enfermedades del Sistema Inmune	8	+ +	->
17	Enfermedades Urogenitales Femeninas y Complicaciones del Embarazo	8	+ +	->
18	Infecciones Bacterianas y Micosis	7	+ +	->
19	Enfermedades Urogenitales Masculinas	5	+ +	->
20	Enfermedades del Sistema Endocrino	5	+ +	->
21	Enfermedades Estomatognáticas	4	+ +	->
22	Virosis	3	+ +	->
23	Enfermedades Parasitarias	1	+ +	->
24	Enfermedades de los Animales	1	+ +	->

Universitat d'Alacant
Universidad de Alicante

Figura E.4: Ejemplo del primer nivel de diagnóstico jerárquico para un caso de ataxia (correspondiente al historial de la figura E.3).

Apéndice F

Consultas a la base de datos PubMed

En las figuras F.1, F.2 y F.3 observamos diversas consultas a la base de datos PubMed, cuyo propósito es recuperar artículos médicos en las siguientes áreas de aplicación:

- **Figura F.1**. Artículos clasificados dentro de las categorías de principios activos, y que puedan resultar útiles para la prescripción de medicamentos.
- **Figura F.2**. Artículos relacionados con plantas medicinales y sus posibles usos terapéuticos.
- **Figura F.3**. Artículos incluidos en cualquier categoría del árbol MeSH, para encontrar y clasificar artículos similares de la Wikipedia que puedan estar relacionados con dichas categorías.

Universitat d'Alacant
Universidad de Alicante

En las consultas observaremos la expresión "*[majr]*" (*major topic*), que restringe la búsqueda para recuperar aquellos documentos que estén relacionados principalmente con la categoría que se especifica (o los nodos hijo que dependan de esa categoría). Respecto a "*[sh]*" (*subheadings*), se utiliza para especificar ciertos matices sobre un tema específico. Por ejemplo, la consulta "*Plant Families and Groups[majr] and therapeutic use[sh]*" recupera aquellos documentos que hablen de plantas medicinales y su uso terapéutico.

("Plant Families and Groups"[majr] OR "Inorganic Chemicals"[majr] OR "Organic Chemicals"[majr] OR "Heterocyclic Compounds"[majr] OR "Polycyclic Compounds"[majr] OR "Macromolecular Substances"[majr] OR "Hormones, Hormone Substitutes, and Hormone Antagonists"[majr] OR "Enzymes and Coenzymes"[majr] OR "Carbohydrates" OR "Lipids"[majr] OR "Amino Acids, Peptides, and Proteins"[majr] OR "Nucleic Acids, Nucleotides, and Nucleosides"[majr] OR "Complex Mixtures"[majr]) AND "therapeutic use"[sh] NOT ("adverse effects"[sh] OR "contraindications"[sh] OR "poisoning"[sh] OR "radiation effects"[sh] OR "toxicity"[sh])

Figura F.1: Consulta a la base de datos PubMed para recuperar artículos clasificados dentro de las categorías de principios activos y que no traten de sus efectos secundarios.

("Plant Families and Groups"[mh] NOT ("adverse effects"[sh] OR "contraindications"[sh] OR "poisoning"[sh] OR "radiation effects"[sh] OR "toxicity"[sh])

Figura F.2: Consulta a la base de datos PubMed para recuperar artículos clasificados dentro de la categoría de plantas medicinales y que no traten sobre sus efectos secundarios.

Universitat d'Alacant
Universidad de Alicante

"Anatomy category"[majr] OR "Organisms category"[majr] OR "Diseases category"[majr] OR "Chemicals and Drugs category"[majr] OR "Analytical, Diagnostic and Therapeutic Techniques and Equipment category"[majr] OR "Psychiatry and Psychology category"[majr] OR "Phenomena and Processes category"[majr] OR "Disciplines and Occupations category"[majr] OR "Anthropology, Education, Sociology and Social Phenomena category"[majr] OR "Technology, Industry, Agriculture category"[majr] OR "Humanities category"[majr] OR "Information Science category"[majr] OR "Named Groups category"[majr] OR "Health Care category"[majr] OR "Publication Characteristics category"[majr] OR "Geographicals category"[majr]

Figura F.3: Consulta a la base de datos PubMed para recuperar artículos clasificados dentro de cualquier categoría del árbol MeSH.

Apéndice G

Aplicaciones Web: estadísticas y comparativas de uso

A continuación se muestran algunas gráficas con estadísticas de acceso por ubicación y comparaciones de uso para las páginas web www.lookfordiagnosis.com, www.lookfortherapy.com[1] y www.lookformedical.com. Para las dos primeras se utilizarán los datos comprendidos en un periodo de tres meses (de septiembre a noviembre de 2009), y para la última, dos meses (octubre y noviembre de 2009). Las resultados se han obtenido por medio de *Google Analytics* y las comparaciones se han realizado tomando como ejemplo páginas web de tamaño y contenidos similares.[2]

En los gráficos contenidos en las figuras G.1, G.2 y G.3 se observa la aceptación que han tenido dichas aplicaciones por parte de los usuarios. De un total de 18 indicadores, sólo 4 son inferiores a la media.

Para lookfordiagnosis.com se observan en la actualidad una media de más de 2.000.000 de visitas únicas mensuales (un 620 % por encima de otras páginas). De estas visitas, un total de 1.651.025 accedieron a la página para realizar consultas sobre posibles diagnósticos.

Para el caso de lookformedical.com, todos los indicadores están por encima de la media.

En los gráficos de las figuras G.4, G.5 y G.6 se observa la diversidad en los idiomas y países de acceso. La cantidad de accesos provinientes de países de habla no inglesa demuestra la importancia de concentrar esfuerzos en ofrecer un entorno multilingüe.

[1] Respecto al dominio lookfortherapy.com, hemos movido toda su funcionalidad a lookfordiagnosis.com, y en la actualidad ya no se encuentra operativo.

[2] Desafortunadamente, Google ha dejado de ofrecer estos servicios, y no podemos proporcionar datos más actualizados.

Figura G.1: Comparativa de uso de `lookfordiagnosis.com`.

Figura G.2: Comparativa de uso de `lookfortherapy.com`.

Figura G.3: Comparativa de uso de `lookformedical.com`.

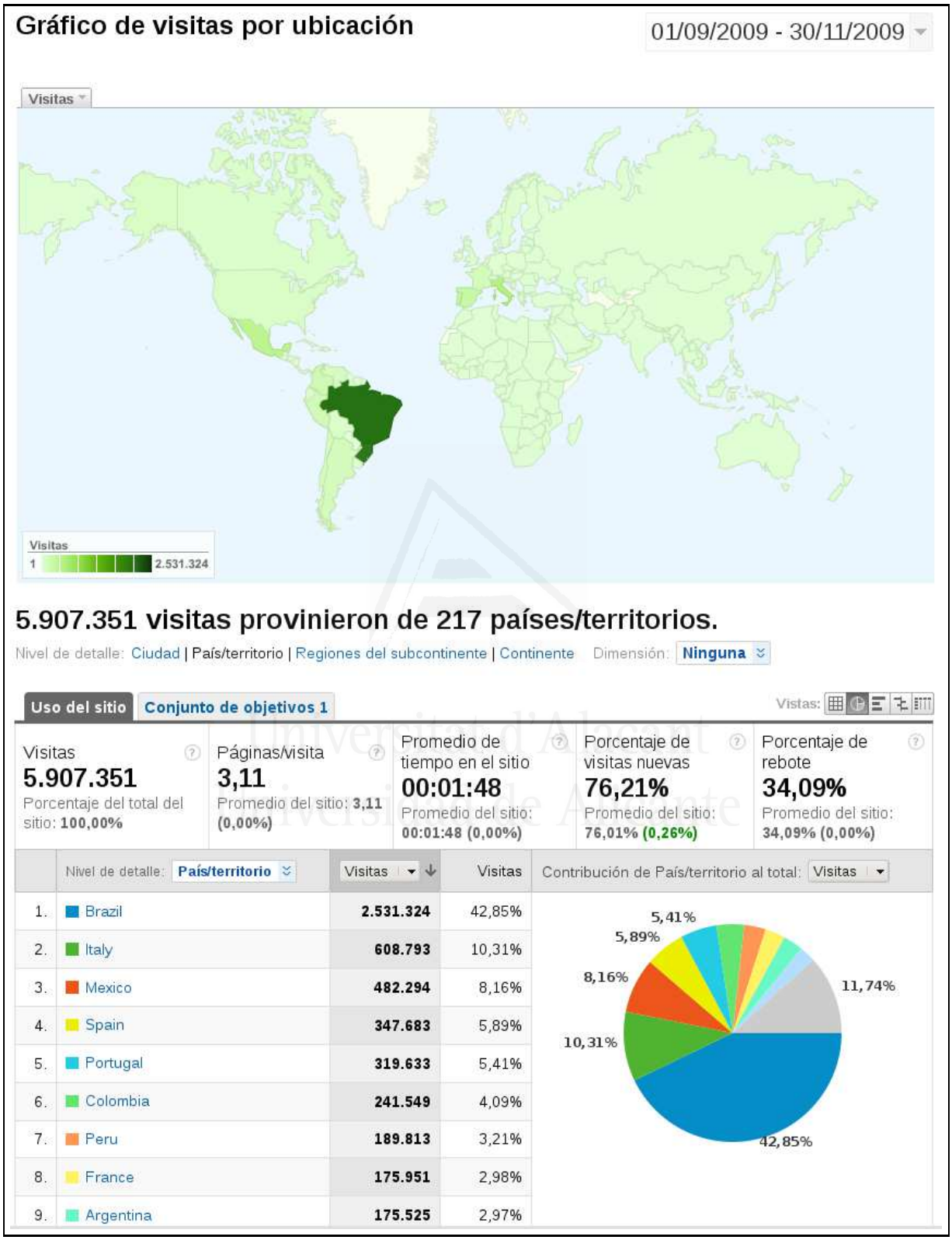


Figura G.4: Visitas de `lookfordiagnosis.com` por ubicación.

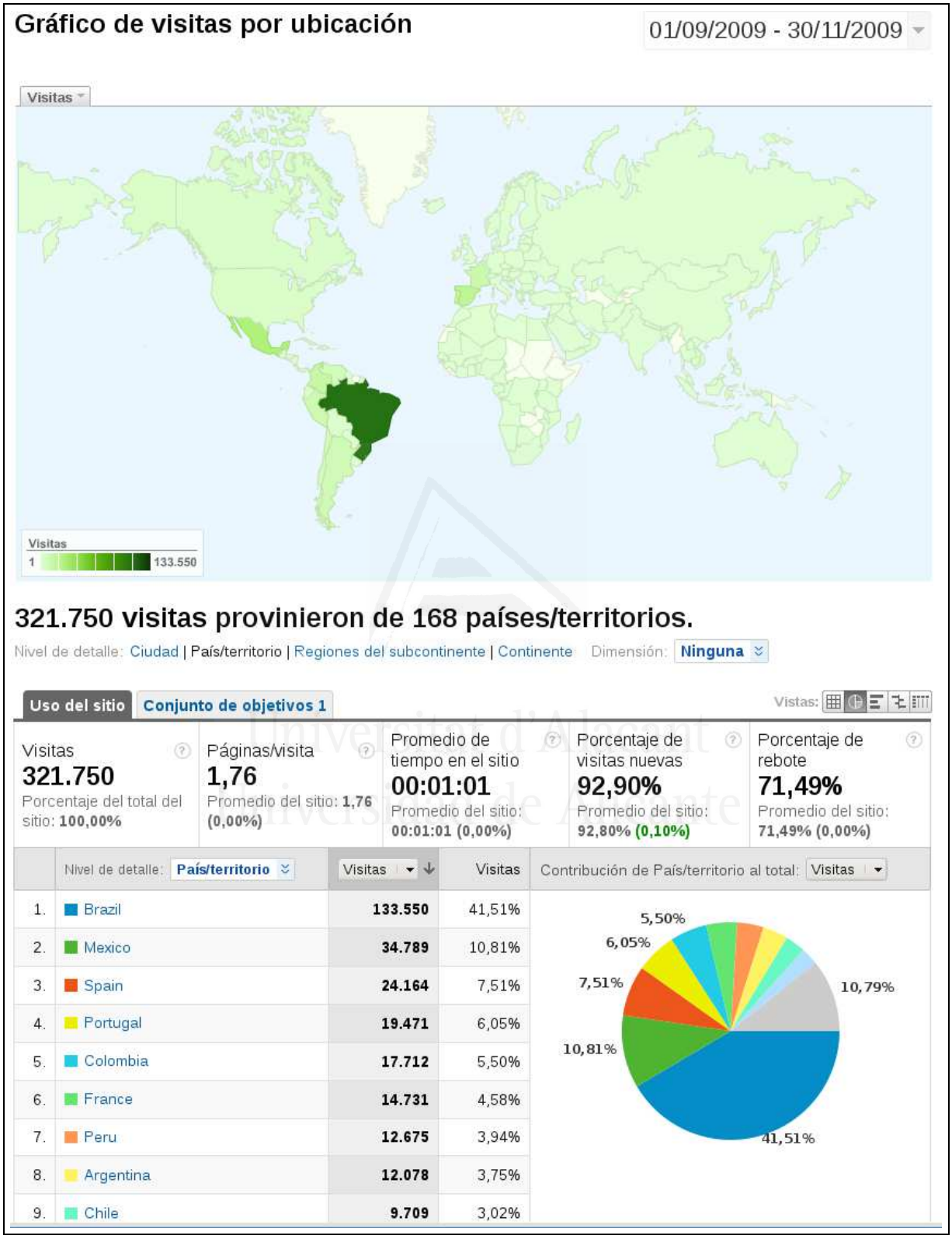


Figura G.5: Visitas de `lookfortherapy.com` por ubicación.

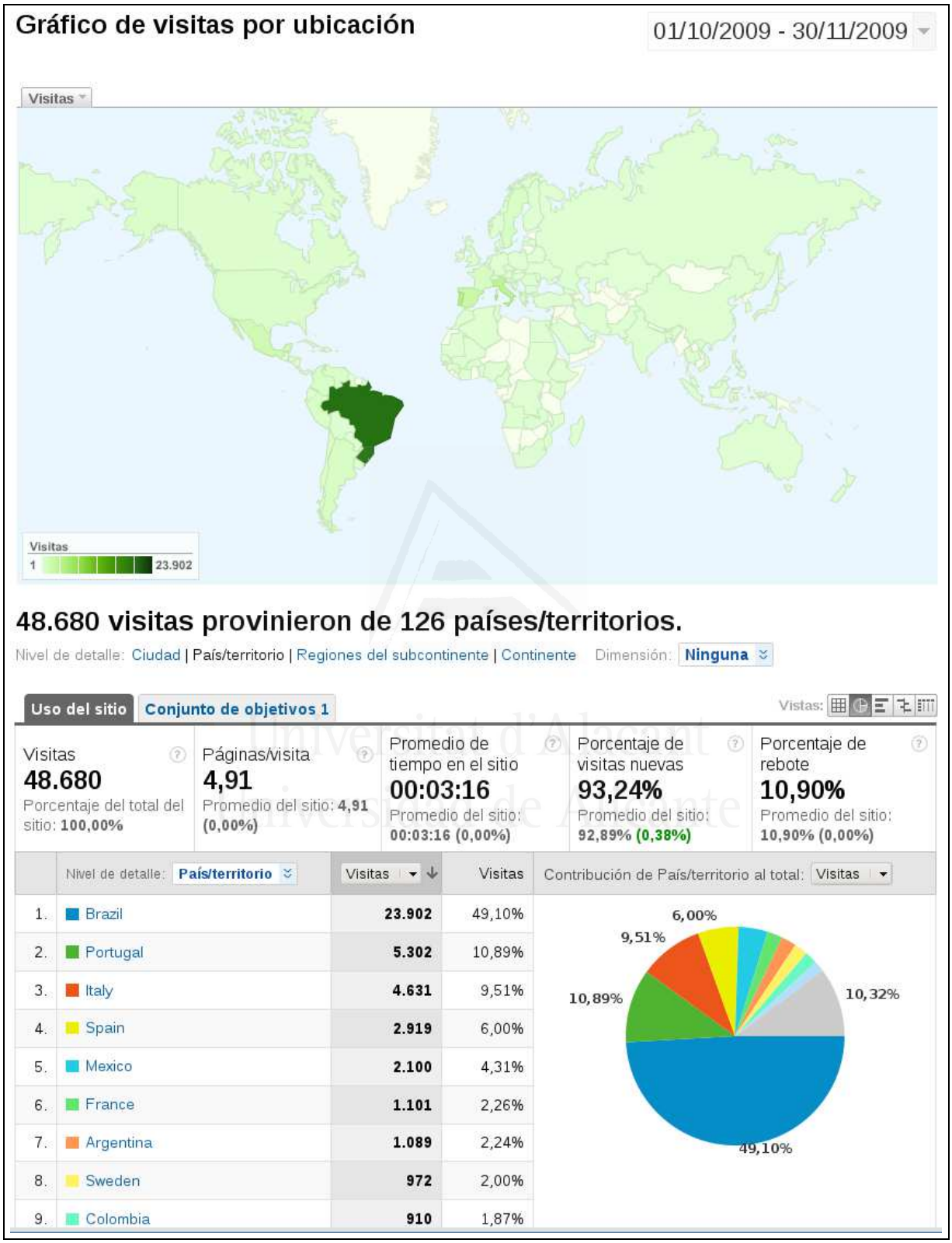


Figura G.6: Visitas de `lookformedical.com` por ubicación.

Universitat d'Alacant
Universidad de Alicante

Bibliografía

Giordano Adami, Paolo Avesani, y Diego Sona. Bootstrapping for hierarchical document classification. In *Proceedings of CIKM-03, 12th ACM International Conference on Information and Knowledge Management*, pages 295–302, New Orleans, US, 2003. ACM Press, New York, US. URL http://doi.acm.org/10.1145/956863.956920.

C.C. Aggarwal y C.X. Zhai. A survey of text classification algorithms. *Mining Text Data*, pages 163–222, 2012.

Erin L. Allwein, Robert E. Schapire, y Yoram Singer. Reducing multiclass to binary: a unifying approach for margin classifiers. *J. Machine Learning Research*, 1:113–141, sep 2001. ISSN 1532-4435. doi: 10.1162/15324430152733133. URL http://dx.doi.org/10.1162/15324430152733133.

Yonatan Amit, Ofer Dekel, y Yoram Singer. A boosting algorithm for label covering in multilabel problems, 2007.

Ion Androutsopoulos, John Koutsias, Konstandinos V. Chandrinos, y Constantine D. Spyropoulos. An experimental comparison of naive bayesian and keyword-based anti-spam filtering with personal e-mail messages. In Nicholas J. Belkin, Peter Ingwersen, y Mun-Kew Leong, editors, *Proceedings of SIGIR-00, 23rd ACM International Conference on Research and Development in Information Retrieval*, pages 160–167, Athens, GR, 2000. ACM Press, New York, US. URL http://doi.acm.org/10.1145/345508.345569.

M. Antonie. Text document categorization by term association. *IEEE International Conference on Data Mining (ICDM)*, pages 19–26, 2002. URL citeseer.ist.psu.edu/620530.html.

Chidanand Apté, Fred J. Damerau, y Sholom M. Weiss. Automated learning of decision rules for text categorization. *ACM Transactions on Information Systems*, 12(3):233–251, 1994a. URL http://www.acm.org/pubs/articles/journals/tois/1994-12-3/p233-apte/p233-apte.pdf.

Chidanand Apté, Fred J. Damerau, y Sholom M. Weiss. Towards language-independent automated learning of text categorization models. In W. Bruce Croft y Cornelis J. Van Rijsbergen, editors, *Proceedings of SIGIR-94, 17th ACM International Conference on Research and*

Universitat d'Alacant

Development in Information Retrieval, pages 23–30, Dublin, IE, 1994b. Springer Verlag, Heidelberg, DE. URL http://www.acm.org/pubs/articles/proceedings/ir/188490/p23-apte/p23-apte.pdf. An extended version appears as Apté et al. [1994a].

L. Douglas Baker y Andrew K. McCallum. Distributional clustering of words for text classification. In W. Bruce Croft, Alistair Moffat, Cornelis J. Van Rijsbergen, Ross Wilkinson, y Justin Zobel, editors, *Proceedings of SIGIR-98, 21st ACM International Conference on Research and Development in Information Retrieval*, pages 96–103, Melbourne, AU, 1998. ACM Press, New York, US. URL http://www.cs.cmu.edu/~mccallum/papers/clustering-sigir98.ps.gz.

Roberto Basili, Alessandro Moschitti, y Maria T. Pazienza. Language-sensitive text classification. In *Proceeding of RIAO-00, 6th International Conference "Recherche d'Information Assistee par Ordinateur"*, pages 331–343, Paris, FR, 2000.

Steven M. Beitzel, Eric C. Jensen, David D. Lewis, Abdur Chowdhury, y Ophir Frieder. Automatic classification of web queries using very large unlabeled query logs. *ACM Trans. Inf. Syst.*, 25(2), april 2007. ISSN 1046-8188. doi: 10.1145/1229179.1229183. URL http://doi.acm.org/10.1145/1229179.1229183.

Ron Bekkerman, Ran El-Yaniv, Naftali Tishby, y Yoad Winter. On feature distributional clustering for text categorization. In W. Bruce Croft, David J. Harper, Donald H. Kraft, y Justin Zobel, editors, *Proceedings of SIGIR-01, 24th ACM International Conference on Research and Development in Information Retrieval*, pages 146–153, New Orleans, US, 2001. ACM Press, New York, US. URL http://www.cs.huji.ac.il/labs/learning/Papers/sigir.ps.gz.

Nicholas J. Belkin y W. Bruce Croft. Information filtering and information retrieval: two sides of the same coin? *Commun. ACM*, 35(12):29–38, 1992. ISSN 0001-0782. doi: http://doi.acm.org/10.1145/138859.138861.

Mohamed Benkhalifa, Amine Bensaid, y Abdelhak Mouradi. Text categorization using the semi-supervised fuzzy c-means algorithm. In *Proceedings of NAFIPS-99, 18th International Conference of the North American Fuzzy Information Processing Society*, pages 561–565, New York, US, 1999.

Alexander Bergo. Text categorization and prototypes. 2002.

Plaban Kumar Bhowmick, Anupam Basu, y Pabitra Mitra. Classifying emotion in news sentences: When machine classification meets human classification, 2010.

Daniel Billsus y Michael Pazzani. A hybrid user model for news stories classification. In *UM99 – Proceedings of the Seventh International Conference on User Modeling*, pages 99–108, Banff, Canada, 1999.

Yevgen Borodin, Valentin Polishchuk, Jalal Mahmud, IV Ramakrishnan, y Amanda Stent. Live and learn from mistakes: A lightweight system for document classification. *Information Processing & Management*, 49(1):83–98, 2013.

Universitat d'Alacant
Universidad de Alicante

Martin Braschler y Bärbel Ripplinger. How effective is stemming and decompounding for German text retrieval? *Information Retrieval*, 7(3/4):291–316, 2004. ISSN 1386-4564. URL http://www.kluweronline.com/issn/1386-4564.

Michael Brückner, Peter Haider, y Tobias Scheffer. Highly scalable discriminative spam filtering. In *TREC*, 2006.

Chris Buckley. The importance of proper weighting methods. *In M. Bates, editor, Human Language Technology Morgan Kaufman*, 1993.

L.S.P. Busagala, W. Ohyama, T. Wakabayashi, y F. Kimura. Machine learning with transformed features in automatic text classification. In *Proceedings of ECML/PKDD-05 Workshop on Subsymbolic Paradigms for Learning in Structured Domains (Relational Machine Learning)*, pages 11–20, 2005.

Lijuan Cai y Thomas Hofmann. Text categorization by boosting automatically extracted concepts. In Jamie Callan, Gordon Cormack, Charles Clarke, David Hawking, y Alan Smeaton, editors, *Proceedings of SIGIR-03, 26th ACM International Conference on Research and Development in Information Retrieval*, pages 182–189, Toronto, CA, 2003. ACM Press, New York, US. URL http://doi.acm.org/10.1145/860435.860470.

Pável Calado, Marco Cristo, Edleno Silva De Moura, Nivio Ziviani, Berthier A. Ribeiro-Neto, y Marcos André Gonçalves. Combining link-based and content-based methods for web document classification. In *Proceedings of CIKM-03, 12th ACM International Conference on Information and Knowledge Management*, pages 394–401, New Orleans, US, 2003. ACM Press, New York, US. URL http://doi.acm.org/10.1145/956863.956938.

Ana Cardoso-Cachopo y Arlindo L. Oliveira. An empirical comparison of text categorization methods. In Mario A. Nascimento, Edleno S. De Moura, y Arlindo L. Oliveira, editors, *Proceedings of SPIRE-03, 10th International Symposium on String Processing and Information Retrieval*, pages 183–196, Manaus, BR, 2003. Springer Verlag, Heidelberg, DE. URL http://www.gia.ist.utl.pt/~acardoso/spire03.pdf. Published in the "Lecture Notes in Computer Science" series, number 2857.

Maria Fernanda Caropreso, Stan Matwin, y Fabrizio Sebastiani. A learner-independent evaluation of the usefulness of statistical phrases for automated text categorization. In Amita G. Chin, editor, *Text Databases and Document Management: Theory and Practice*, pages 78–102. Idea Group Publishing, Hershey, US, 2001. URL http://www.math.unipd.it/~fabseb60/Publications/TD01a.pdf.

Marine Carpuat, Weifeng Su, y Dekai Wu. Augmenting ensemble classification for word sense disambiguation with a kernel pca model. In *In Proceedings of Senseval-3, Third International Workshop on Evaluating Word Sense Disambiguation Systems*, pages 88–92, 2004.

Enrique Castillo, José Manuel Gutiérrez, y Ali S. Hadi. *Sistemas expertos y modelos de redes probabilísticos*. Academia de Ingeniería, Madrid, 1996.

William B. Cavnar y John M. Trenkle. N-gram-based text categorization. In *Proceedings of SDAIR-94, 3rd Annual Symposium on Document Analysis and Information Retrieval*, pages 161–175, Las Vegas, US, 1994. URL http://www.nonlineardynamics.com/trenkle/papers/sdair-94-bc.ps.gz.

Michelangelo Ceci y Donato Malerba. Classifying web documents in a hierarchy of categories: a comprehensive study. *J. Intell. Inf. Syst.*, 28(1):37–78, 2007. ISSN 0925-9902. doi: http://dx.doi.org/10.1007/s10844-006-0003-2.

Miquel Centelles. Taxonomías para la categorización y la organización de la información en sitios web. *Hipertext.net*, 3, 2005. URL http://www.hipertext.net/web/pag264.htm.

Chih-Chung Chang y Chih-Jen Lin. *LIBSVM: a library for support vector machines*, 2001. Software available at *http://www.csie.ntu.edu.tw/ cjlin/libsvm*.

Olivier Chapelle, Bernhard Schölkopf, y Alexander Zien, editors. *Semi-Supervised Learning*. MIT Press, Cambridge, MA, 2006.

Chien Chin Chen, Meng Chang Chen, y Yeali Sun. Pva: A self-adaptive personal view agent. *Journal of Intelligent Information Systems*, 18(2/3):173–194, 2002. URL http://www.wkap.nl/article.pdf?391245. Special Issue on Automated Text Categorization.

Universidad Chile. Filtraje del spam usando spamassassin. Technical report, Universidad Técnica Federico Santa María (Chile). Dirección de Tecnologías de la Información, 2010. Accesible desde http://www.dcsc.utfsm.cl/redes/email/spam/.

William W. Cohen y Yoram Singer. Context-sensitive learning methods for text categorization. In Hans-Peter Frei, Donna Harman, Peter Schäuble, y Ross Wilkinson, editors, *Proceedings of SIGIR-96, 19th ACM International Conference on Research and Development in Information Retrieval*, pages 307–315, Zürich, CH, 1996. ACM Press, New York, US. URL http://www.research.whizbang.com/~wcohen/postscript/sigir-96.ps. An extended version appears as Cohen y Singer [1999].

William W. Cohen y Yoram Singer. Context-sensitive learning methods for text categorization. *ACM Transactions on Information Systems*, 17(2):141–173, 1999. URL http://www.acm.org/pubs/articles/journals/tois/1999-17-2/p141-cohen/p141-cohen.pdf.

Corinna Cortes y Vladimir Vapnik. Support-vector networks. *Machine Learning*, 20(3):273–297, September 1995. ISSN 0885-6125. doi: 10.1007/BF00994018. URL http://dx.doi.org/10.1007/BF00994018.

Cristóobal Camarero Coterillo. An adaptation of soucy and mineau weightin for large scale text classification, 2010.

Koby Crammer y Yoram Singer. A new family of online algorithms for category ranking. In Micheline Beaulieu, Ricardo Baeza-Yates, Sung Hyon Myaeng, y Kalervo Järvelin, editors,

Proceedings of SIGIR-02, 25th ACM International Conference on Research and Development in Information Retrieval, pages 151–158, Tampere, FI, 2002. ACM Press, New York, US. URL http://doi.acm.org/10.1145/564376.564404.

Koby Crammer y Yoram Singer. A family of additive online algorithms for category ranking. *Journal of Machine Learning Research*, 2003.

MAG Cumbreras, LAU Lpez, FM Santiago, y AM Raez. Búsqueda de respuestas multilingüe: Clasificación de preguntas en español basada en aprendizaje. 2005. URL http://www.sepln.org/revistaSEPLN/revista/34/03.pdf.

James R. Curran y Marc Moens. Scaling context space. In *In Proceedings of the 40th annual meeting of the Association for Computational Linguistics*, pages 231–238, 2002.

M. Dash y H. Liu. Feature selection for classification. *Intelligent Data Analysis*, 1:131–156, 1997.

Franca Debole y Fabrizio Sebastiani. Supervised term weighting for automated text categorization. In *Proceedings of SAC-03, 18th ACM Symposium on Applied Computing*, pages 784–788, Melbourne, US, 2003. ACM Press, New York, US. URL http://www.math.unipd.it/~fabseb60/Publications/SAC03b.pdf. An extended version appears as Debole y Sebastiani [2004b].

Franca Debole y Fabrizio Sebastiani. An analysis of the relative difficulty of reuters-21578 subsets. In *Proceedings of LREC-04, 4th International Conference on Language Resources and Evaluation*, Lisbon, PT, 2004a. URL http://www.math.unipd.it/~fabseb60/Publications/LREC04.pdf. Forthcoming.

Franca Debole y Fabrizio Sebastiani. Supervised term weighting for automated text categorization. In Spiros Sirmakessis, editor, *Text Mining and its Applications*, Number 138 in the "Studies in Fuzziness and Soft Computing" series, pages 81–98. Physica-Verlag, Heidelberg, DE, 2004b. URL http://www.math.unipd.it/~fabseb60/Publications/NEMIS04.pdf.

Franca Debole y Fabrizio Sebastiani. An analysis of the relative hardness of reuters-21578 subsets. In *Proceedings of LREC-04, 4th International Conference on Language Resources and Evaluation*, pages 971–974, Lisbon, PT, 2004c. URL http://www.math.unipd.it/~fabseb60/Publications/LREC04.pdf.

Inderjit Dhillon, Subramanyam Mallela, y Rahul Kumar. A divisive information-theoretic feature clustering algorithm for text classification. *Journal of Machine Learning Research*, 3: 1265–1287, March 2003. URL http://www.jmlr.org/papers/volume3/dhillon03a/dhillon03a.pdf.

Giorgio M. Di Nunzio y Alessandro Micarelli. Does a new simple gaussian weighting approach perform well in text categorization? In Georg Gottlob y Toby Walsh, editors, *Proceedings of IJCAI-03, 18th International Joint Conference on Artificial Intelligence*, pages 581–586, Acapulco, MX, 2003. Morgan Kaufmann Publishers, San Francisco, US.

Yanlei Diao, Hongjun Lu, y Dekai Wu. A comparative study of classification-based personal e-mail filtering. In Takao Terano, Huan Liu, y Arbee L.P. Chen, editors, *Proceedings of PAKDD-00, 4th Pacific-Asia Conference on Knowledge Discovery and Data Mining*, pages 408–419, Kyoto, JP, 2000. Springer Verlag, Heidelberg, DE. URL http://www.cs.berkeley.edu/~diaoyl/publications/pakdd00.ps. Published in the "Lecture Notes in Computer Science" series, number 1805.

A. Dörfler, S. Eilert, A. Mentrup, M. E. Müller, R. Rolf, C. R. Rollinger, F. Sievertsen, y F. Trenkamp. Bikini: User adaptive news classification in the world wide web, 2001. URL http://citeseerx.ist.psu.edu/viewdoc/summary?doi=10.1.1.28.6338.

Harris Drucker, Vladimir Vapnik, y Dongui Wu. Support vector machines for spam categorization. *IEEE Transactions on Neural Networks*, 10(5):1048–1054, 1999. URL http://www.monmouth.edu/~drucker/SVM_spam_article_compete.PDF.

Richard O. Duda y Peter E. Hart. *Pattern Classification and Scene Analysis*. John Wiley & Sons Inc, 1973. ISBN 0471223611. URL http://www.amazon.com/exec/obidos/redirect?tag=citeulike07-20&path=ASIN/0471223611.

S.T. Dumais. Improving the retrieval of information from external sources, 1991.

Susan T. Dumais y Hao Chen. Hierarchical classification of web content. In Nicholas J. Belkin, Peter Ingwersen, y Mun-Kew Leong, editors, *Proceedings of SIGIR-00, 23rd ACM International Conference on Research and Development in Information Retrieval*, pages 256–263, Athens, GR, 2000. ACM Press, New York, US. URL http://research.microsoft.com/~sdumais/sigir00.pdf.

Susan T. Dumais, John Platt, David Heckerman, y Mehran Sahami. Inductive learning algorithms and representations for text categorization. In Georges Gardarin, James C. French, Niki Pissinou, Kia Makki, y Luc Bouganim, editors, *Proceedings of CIKM-98, 7th ACM International Conference on Information and Knowledge Management*, pages 148–155, Bethesda, US, 1998. ACM Press, New York, US. URL http://robotics.stanford.edu/users/sahami/papers-dir/cikm98.pdf.

Ted Dunning. Accurate methods for the statistics of surprise and coincidence. *Comput. Linguist.*, 19(1):61–74, 1993. ISSN 0891-2017.

EAGLES. Corpus typology: a framework for classification, 1996.

Ran El-Yaniv y Oren Souroujon. Iterative double clustering for unsupervised and semi-supervised learning. In Luc De Raedt y Peter A. Flach, editors, *Proceedings of ECML-01, 12th European Conference on Machine Learning*, pages 121–132, Freiburg, DE, 2001. Springer Verlag, Heidelberg, DE. URL http://link.springer.de/link/service/series/0558/papers/2167/21670121.pdf. Published in the "Lecture Notes in Computer Science" series, number 2167.

Gerard Escudero, Lluís Màrquez, y German Rigau. Boosting applied to word sense disambiguation. In Ramon López De Mántaras y Enric Plaza, editors, *Proceedings of ECML-00, 11th European Conference on Machine Learning*, pages 129–141, Barcelona, ES, 2000. Springer Verlag, Heidelberg, DE. URL http://www.lsi.upc.es/~escudero/recerca/ecml00.pdf. Published in the "Lecture Notes in Computer Science" series, number 1810.

Carlos G. Figuerola, Angel Francisco, Zazo Rodrguez, Jos Luis, Alonso Berrocal, y Facultad De Documentacin. Automatic vs. manual categorisation of documents in spanish, 2001.

Michelle Fisher y Richard Everson. When are links useful? experiments in text classification. In Fabrizio Sebastiani, editor, *Proceedings of ECIR-03, 25th European Conference on Information Retrieval*, pages 41–56, Pisa, IT, 2003. Springer Verlag. URL http://link.springer.de/link/service/series/0558/papers/2633/26330041.pdf.

Imola Fodor. A survey of dimension reduction techniques, 2002. URL http://citeseerx.ist.psu.edu/viewdoc/summary?doi=10.1.1.8.5098.

George Forman. Choose your words carefully: An empirical study of feature selection metrics for text classification. In Tapio Elomaa, Heikki Mannila, y H. Toivonen, editors, *Proceedings of PKDD-02, 6th European Conference on Principles of Data Mining and Knowledge Discovery*, pages 150–162, Helsinki, FI, 2002. Springer Verlag, Heidelberg, DE. URL http://springerlink.metapress.com/openurl.asp?genre=article&issn=0302-9743&volume=2431&spage=150. Published in the "Lecture Notes in Computer Science" series, number 2431.

George Forman. An extensive empirical study of feature selection metrics for text classification. *Journal of Machine Learning Research*, 3:1289–1305, March 2003. URL http://www.jmlr.org/papers/v3/forman03a.html.

George Forman. Bns feature scaling: an improved representation over tf-idf for svm text classification. In *CIKM '08: Proceeding of the 17th ACM conference on Information and knowledge mining*, pages 263–270, New York, NY, USA, 2008. ACM. ISBN 978-1-59593-991-3. doi: 10.1145/1458082.1458119. URL http://dx.doi.org/10.1145/1458082.1458119.

C. Fox. Lexical analysis and stoplists. In W. B. Frakes y Baeza R. Yates, editors, *Information Retrieval: Data Structures and Algorithms*. Englewood Cliffs, NJ: Prentice Hall, 1992.

W.B. Frakes y R. Baeza-Yates. *Information Retrieval: Data, Structures and Algorithms*. Prentice Hall, 1992.

J. Furnkranz. A study using n-gram features for text categorization, 1998.

William A. Gale, Kenneth W. Church, y David Yarowsky. A method for disambiguating word senses in a large corpus. *Computers and the Humanities*, 26(5):415–439, 1993. URL http://www.research.att.com/~kwc/published_1993_sense.ps.

Universidad de Alicante

Ashutosh Garg y Dan Roth. Understanding probabilistic classifiers. In *EMCL '01: Proceedings of the 12th European Conference on Machine Learning*, pages 179–191, London, UK, 2001. Springer-Verlag. ISBN 3-540-42536-5.

Susan Gauch, Aravind Ch, y Shankar Ranganathan. Training a hierarchical classifier using inter-document relationships, 2006.

Tanja Gaustad, Gosse Bouma, y Rijksuniversiteit Groningen. Accurate stemming of dutch for text classification. In *Language and Computers*, pages 104–117, 2002.

Ignacio Giraldez, Enrique Puertas, Jose Maria Gomez, Raul Murciano, Inmaculada Chacon, y Dpto Periodismo Especializado. Hermes: Intelligent multilingual news filtering based on language engineering for advanced user profiling, 2002.

Andrew R. Golding y Yves Schabes. Combining trigram-based and feature-based methods for context-sensitive spelling correction. In *Proceedings of the 34th annual meeting on Association for Computational Linguistics*, pages 71–78, Morristown, NJ, USA, 1996. Association for Computational Linguistics. doi: http://dx.doi.org/10.3115/981863.981873.

José M. Gómez, Enrique Puertas, Manuel Buenaga, y Francisco Carrero. Text filtering at poesia: a new internet content filtering tool for educational environments. 2002. URL www.esi.uem.es/~jmgomez/papers/sepln02.pdf.

José M. Gómez, Ignacio Giráldez, y Manuel Buenaga. Text categorization for internet content filtering. 2004. URL http://tornado.dia.fi.upm.es/caepia/numeros/22/raepiaF09.pdf.

Michael Granitzer. Hierarchical text classification using methods from machine learning. Master's thesis, Graz University of Technology, 2003.

Gongde Guo, Hui Wang, David A. Bell, Yaxin Bi, y Kieran Greer. An knn model-based approach and its application in text categorization. In Alexander F. Gelbukh, editor, *Proceedings of CICLING-04, 5th International Conference on Computational Linguistics and Intelligent Text Processing*, pages 559–570, Seoul, KO, 2004. Springer Verlag, Heidelberg, DE. Published in the "Lecture Notes in Computer Science" series, number 2945.

Louise Guthrie, Joe A. Guthrie, y James Leistensnider. Document classification and routing. In Tomek Strzalkowski, editor, *Natural language information retrieval*, pages 289–310. Kluwer Academic Publishers, Dordrecht, NL, 1999.

Xuexian Han, Guowei Zu, Wataru Ohyama, Tetsushi Wakabayashi, y Fumitaka Kimura. Accuracy improvement of automatic text classification based on feature transformation and multi-classifier combination. In *AWCC*, pages 463–468, 2004.

Samer Hassan y Carmen Banea. Random-walk term weighting for improved text classification. In *TextGraphs '06: Proceedings of TextGraphs: the First Workshop on Graph Based Methods for Natural Language Processing on the First Workshop on Graph Based Methods*

for Natural Language Processing, pages 53–60, Morristown, NJ, USA, 2006. Association for Computational Linguistics.

Philip J. Hayes y Steven P. Weinstein. CONSTRUE/TIS: a system for content-based indexing of a database of news stories. In Alain Rappaport y Reid Smith, editors, *Proceedings of IAAI-90, 2nd Conference on Innovative Applications of Artificial Intelligence*, pages 49–66, Boston, US, 1990. AAAI Press, Menlo Park, US.

William Hersh, Christopher Buckley, T.J. Leone, y David Hickman. OHSUMED: an interactive retrieval evaluation and new large text collection for research. In W. Bruce Croft y Cornelis J. Van Rijsbergen, editors, *Proceedings of SIGIR-94, 17th ACM International Conference on Research and Development in Information Retrieval*, pages 192–201, Dublin, IE, 1994. Springer Verlag, Heidelberg, DE. URL http://www.acm.org/pubs/articles/proceedings/ir/188490/p192-hersh/p192-hersh.pdf.

C.W. Hsu y C.J. Lin. A comparison of methods for multiclass support vector machines. *Neural Networks, IEEE Transactions on*, 13(2):415–425, 2002.

Zhiheng Huang, Marcus Thint, y Asli Celikyilmaz. Investigation of question classifier in question answering. In *Proceedings of the 2009 Conference on Empirical Methods in Natural Language Processing: Volume 2*, EMNLP '09, pages 543–550, Stroudsburg, PA, USA, 2009. Association for Computational Linguistics. ISBN 978-1-932432-62-6. URL http://dl.acm.org/citation.cfm?id=1699571.1699583.

Collier Nigel Ibushi, Katsutoshi y Jun'ichi Tsujii. Classification of medline abstracts. *Genome Informatics, volume 10*, pages 290–291, 1999.

José Sánchez Jiménez. La documentación en el proceso de evaluación de sistemas de clasificación automática. *Documentación de las Ciencias de la Información*, 30:25–44, 2007.

Thorsten Joachims. A probabilistic analysis of the rocchio algorithm with tfidf for text categorization. In Douglas H. Fisher, editor, *Proceedings of ICML-97, 14th International Conference on Machine Learning*, pages 143–151, Nashville, US, 1997. Morgan Kaufmann Publishers, San Francisco, US. URL http://www-ai.cs.uni-dortmund.de/DOKUMENTE/joachims_97a.ps.gz.

Thorsten Joachims. Text categorization with support vector machines: learning with many relevant features. In Claire Nédellec y Céline Rouveirol, editors, *Proceedings of ECML-98, 10th European Conference on Machine Learning*, pages 137–142, Chemnitz, DE, 1998. Springer Verlag, Heidelberg, DE. URL http://www-ai.cs.uni-dortmund.de/DOKUMENTE/joachims_98a.ps.gz. Published in the "Lecture Notes in Computer Science" series, number 1398.

Thorsten Joachims. *Making large-Scale SVM Learning Practical. Advances in Kernel Methods - Support Vector Learning.* 1999. Software available at *http://svmlight.joachims.org/*.

Universitat d'Alacant
Universidad de Alicante

Thorsten Joachims. Estimating the generalization performance of a svm efficiently. In Pat Langley, editor, *Proceedings of ICML-00, 17th International Conference on Machine Learning*, pages 431–438, Stanford, US, 2000. Morgan Kaufmann Publishers, San Francisco, US. URL http://www-ai.cs.uni-dortmund.de/DOKUMENTE/joachims_00a.pdf.

Thorsten Joachims. *Learning to Classify Text using Support Vector Machines*. Kluwer Academic Publishers, Dordrecht, NL, 2002.

Thorsten Joachims. *Support Vector and Kernel Methods*, 2003. SIGIR 2003 Tutorial.

Thorsten Joachims. Training linear svms in linear time. In *KDD '06: Proceedings of the 12th ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 217–226, New York, NY, USA, 2006. ACM. ISBN 1-59593-339-5. doi: 10.1145/1150402.1150429. URL http://dx.doi.org/10.1145/1150402.1150429.

T. Kalt. A new probabilistic model of text classification and retrieval. Technical Report UM-CS-1998-018, , 1998. URL http://citeseer.ist.psu.edu/273666.html.

Jun Karamon, Yutaka Matsuo, Hikaru Yamamoto, y Mitsuru Ishizuka. Generating social network features for link-based classification. In Joost N. Kok, Jacek Koronacki, Ramon L. de Mántaras, Stan Matwin, Dunja Mladenic, y Andrzej Skowron, editors, *PKDD*, volume 4702 of *Lecture Notes in Computer Science*, pages 127–139. Springer, 2007.

Adam Kilgarriff. What is word sense disambiguation good for? *CoRR*, cmp-lg/9712008, 1997.

Yu-Hwan Kim, Shang-Yoon Hahn, y Byoung-Tak Zhang. Text filtering by boosting naive bayes classifiers. In Nicholas J. Belkin, Peter Ingwersen, y Mun-Kew Leong, editors, *Proceedings of SIGIR-00, 23rd ACM International Conference on Research and Development in Information Retrieval*, pages 168–175, Athens, GR, 2000. ACM Press, New York, US. URL http://www.acm.org/pubs/articles/proceedings/ir/345508/p168-kim/p168-kim.pdf.

Youngjoong Ko y Jungyun Seo. Automatic text categorization by unsupervised learning. In *Proceedings of COLING-00, the 18th International Conference on Computational Linguistics*, Saarbrücken, DE, 2000. URL http://nlp3.korea.ac.kr/proceeding/coling2000/COLING/ps/066.ps.

Philipp Koehn, Hieu Hoang, Alexandra Birch, Chris Callison-Burch, Marcello Federico, Nicola Bertoldi, Brooke Cowan, Wade Shen, Christine Moran, Richard Zens, et al. Moses: Open source toolkit for statistical machine translation. In *Proceedings of the 45th Annual Meeting of the ACL on Interactive Poster and Demonstration Sessions*, pages 177–180. Association for Computational Linguistics, 2007.

Ron Kohavi y George H. John. Wrappers for feature subset selection. *Artificial Intelligence*, 97(1-2):273–324, 1997. URL http://citeseerx.ist.psu.edu/viewdoc/summary?doi=10.1.1.48.720.

Daphne Koller y Mehran Sahami. Hierarchically classifying documents using very few words. In Douglas H. Fisher, editor, *Proceedings of ICML-97, 14th International Conference on Machine Learning*, pages 170–178, Nashville, US, 1997. Morgan Kaufmann Publishers, San Francisco, US. URL http://robotics.stanford.edu/users/sahami/papers-dir/ml97-hier.ps.

Madhusudhan Kongovi, Juan Carlos Guzman, y Venu Dasigi. Text categorization: An experiment using phrases. In Fabio Crestani, Mark Girolami, y Cornelis J. Van Rijsbergen, editors, *Proceedings of ECIR-02, 24th European Colloquium on Information Retrieval Research*, pages 213–228, Glasgow, UK, 2002. Springer Verlag, Heidelberg, DE. URL http://link.springer.de/link/service/series/0558/papers/2291/22910213.pdf. Published in the "Lecture Notes in Computer Science" series, number 2291.

Raymond Kosala y Hendrik Blockeel. Web mining research: A survey. *SIGKDD Explorations*, 2:1–15, 2000.

M. K. Kozlov, S. P. Tarasov, y L. G. Khachiyan. Polynomial solvability of convex quadratic programming. *Doklady Akademiia Nauk SSSR*, 248, 1979.

Savio L. Lam y Dik L. Lee. Feature reduction for neural network based text categorization. In Arbee L. Chen y Frederick H. Lochovsky, editors, *Proceedings of DASFAA-99, 6th IEEE International Conference on Database Advanced Systems for Advanced Application*, pages 195–202, Hsinchu, TW, 1999. IEEE Computer Society Press, Los Alamitos, US. URL http://dlib.computer.org/conferen/dasfaa/0084/pdf/00840195.pdf.

Man Lan, Chew-Lim Tan, Hwee-Boon Low, y Sam-Yuan Sung. A comprehensive comparative study on term weighting schemes for text categorization with support vector machines. In *WWW '05: Special interest tracks and posters of the 14th international conference on World Wide Web*, pages 1032–1033, New York, NY, USA, 2005. ACM. ISBN 1-59593-051-5. doi: http://doi.acm.org/10.1145/1062745.1062854.

Man Lan, Chew-Lim Tan, y Hwee-Boon Low. Proposing a new term weighting scheme for text categorization. In *AAAI'06: Proceedings of the 21st national conference on Artificial intelligence*, pages 763–768. AAAI Press, 2006. ISBN 978-1-57735-281-5.

Man Lan, Chew Lim Tan, Jian Su, y Yue Lu. Supervised and traditional term weighting methods for automatic text categorization. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 31:721–735, 2009. ISSN 0162-8828. doi: http://doi.ieeecomputersociety.org/10.1109/TPAMI.2008.110.

Ken Lang. NewsWeeder: learning to filter netnews. In Armand Prieditis y Stuart J. Russell, editors, *Proceedings of ICML-95, 12th International Conference on Machine Learning*, pages 331–339, Lake Tahoe, US, 1995. Morgan Kaufmann Publishers, San Francisco, US.

Pat Langley. Elements of machine learning. 1996.

K. W. Lau y Q. H. Wu. Leave one support vector out cross validation for fast estimation of generalization errors. *Pattern Recognition*, 37(9):1835–1840, 2004. URL http://www.sciencedirect.com/science/article/B6V14-4CPCPV7-2/2/8577be3c704087a252047c48e8f2f73d.

Verayuth Lertnattee y Thanaruk Theeramunkong. Effect of term distributions on centroid-based text categorization. *Information Sciences*, 158(1):89–115, 2004. URL http://dx.doi.org/10.1016/j.ins.2003.07.007.

David D. Lewis. Data extraction as text categorization: An experiment with the MUC-3 corpus. In *Proceedings of MUC-3, 3rd Message Understanding Conference*, pages 245–255, San Diego, US, 1991. Morgan Kaufmann Publishers, San Francisco, US. URL http://www.research.att.com/~lewis/papers/lewis91c.ps.

David D. Lewis. An evaluation of phrasal and clustered representations on a text categorization task. In Nicholas J. Belkin, Peter Ingwersen, y Annelise Mark Pejtersen, editors, *Proceedings of SIGIR-92, 15th ACM International Conference on Research and Development in Information Retrieval*, pages 37–50, Kobenhavn, DK, 1992a. ACM Press, New York, US. URL http://www.research.att.com/~lewis/papers/lewis92b.ps.

David D. Lewis. *Representation and learning in information retrieval*. PhD thesis, Department of Computer Science, University of Massachusetts, Amherst, US, 1992b. URL http://www.research.att.com/~lewis/papers/lewis91d.ps.

David D. Lewis. Feature selection and feature extraction for text categorization. In *In Proceedings of Speech and Natural Language Workshop*, pages 212–217. Morgan Kaufmann, 1992c.

David D. Lewis. A sequential algorithm for training text classifiers: corrigendum and additional data. *SIGIR Forum*, 29(2):13–19, 1995. URL http://www.research.att.com/~lewis/papers/lewis95g.ps.

David D. Lewis y William A. Gale. A sequential algorithm for training text classifiers. In W. Bruce Croft y Cornelis J. Van Rijsbergen, editors, *Proceedings of SIGIR-94, 17th ACM International Conference on Research and Development in Information Retrieval*, pages 3–12, Dublin, IE, 1994. Springer Verlag, Heidelberg, DE. URL http://www.research.att.com/~lewis/papers/lewis94c.ps. See also Lewis [1995].

David D. Lewis y Marc Ringuette. A comparison of two learning algorithms for text categorization. In *Proceedings of SDAIR-94, 3rd Annual Symposium on Document Analysis and Information Retrieval*, pages 81–93, Las Vegas, US, 1994. URL http://www.research.att.com/~lewis/papers/lewis94b.ps.

David D. Lewis, Robert E. Schapire, James P. Callan, y Ron Papka. Training algorithms for linear text classifiers. In Hans-Peter Frei, Donna Harman, Peter Schäuble, y Ross Wilkinson, editors, *Proceedings of SIGIR-96, 19th ACM International Conference on Research and*

Development in Information Retrieval, pages 298–306, Zürich, CH, 1996. ACM Press, New York, US. URL http://www.research.att.com/~lewis/papers/lewis96d.ps.

Hang Li y Cong Li. Word translation disambiguation using bilingual bootstrapping. *Comput. Linguist.*, 30(1):1–22, 2004. ISSN 0891-2017. doi: http://dx.doi.org/10.1162/089120104773633367.

Yong H. Li y Anil K. Jain. Classification of text documents. *The Computer Journal*, 41(8): 537–546, 1998.

Liu Li-zhen, Song Han-tao, y Lu Yu-chang. Dimensionality reduction by mutual information in text classification. *Journal of Beijing Institute of Technology*, 14(1):32–36, 2005.

Elizabeth D. Liddy, Woojin Paik, y Edmund S. Yu. Text categorization for multiple users based on semantic features from a machine-readable dictionary. *ACM Transactions on Information Systems*, 12(3):278–295, 1994. URL http://www.acm.org/pubs/articles/journals/tois/1994-12-3/p278-liddy/p278-liddy.pdf.

Ying Liu y Han Tong Loh. A simple probability based term weighting scheme for automated text classification. In *IEA/AIE'07: Proceedings of the 20th international conference on Industrial, engineering, and other applications of applied intelligent systems*, pages 33–43, Berlin, Heidelberg, 2007. Springer-Verlag. ISBN 978-3-540-73322-5.

Ying Liu, Han Loh, Youcef-Toumi Kamal, y Shu Tor. Handling of imbalanced data in text classification: Category-based term weights. In Anne Kao y Stephen R. Poteet, editors, *Natural Language Processing and Text Mining*, pages 171–192. Springer London, 2007. ISBN 978-1-84628-754-1.

Babak Loni. A survey of state-of-the-art methods on question classification. *TU Delft Repository*, 2011. URL http://repository.tudelft.nl/view/ir/uuid:8e57caa8-04fc-4fe2-b668-20767ab3db92/.

Zhihong Lu y Kathryn S. McKinley. Partial collection replication for information retrieval. *Information Retrieval*, 6(2):159–198, 2003. ISSN 1386-4564. URL http://www.kluweronline.com/issn/1386-4564.

H. P. Luhn. The automatic creation of literature abstracts. *IBM Journal of Research and Development*, 2(2), 1958. URL http://domino.watson.ibm.com/tchjr/journalindex.nsf/c469af92ea9eceac85256bd50048567c/97e0420635a5000a85256bfa00683d33?OpenDocument.

Hwee-Boon Low Chew-Lim Tan Man Lan, Sam-Yuan Sung. A comparative study on term weighting schemes for text categorization. In *IEEE International Joint Conference on Neural Networks, 2005. IJCNN '05. Proceedings*, pages 546–551. IEEEXplore, 2005. ISBN 0-7803-9048-2.

Christopher D. Manning, Prabhakar Raghavan, y Hinrich Schtze. *Introduction to Information Retrieval*. Cambridge University Press, New York, NY, USA, 2008. ISBN 0521865719, 9780521865715.

Nuno Maria y Mário J. Silva. Theme-based retrieval of web news. In *SIGIR*, pages 354–356, 2000.

Lluís Màrquez y Jesús Giménez. A general pos tagger generator based on support vector machines. *Journal of Machine Learning Research*, 2004.

Bart Massey, Mick Thomure, Raya Budrevich, y Scott Long. Learning spam: simple techniques for freely-available software. In *Proceedings of the annual conference on USENIX Annual Technical Conference*, ATEC '03, pages 13–13, Berkeley, CA, USA, 2003. USENIX Association. URL http://dl.acm.org/citation.cfm?id=1247340.1247353.

Andrew McCallum y K. Nigam. A comparison of event models for naive bayes text classification. In *Proceedings of AAAI-98, Workshop on Learning for Text Categorization*, 1998. URL citeseer.nj.nec.com/mccallum98comparison.html.

Arkaitz Zubiaga Mendialdua. *Aproximaciones a SVM semisupervisado multiclase para clasificación de páginas web. Memoria de Tesis de Máster*. Departamento de Lenguajes y Sistemas Informáticos. ETS de Ingeniería Informática. Universidad Nacional de Educación a Distancia, 2008.

Dimitris Meretakis y Beat Wuthrich. Extending naive bayes classifiers using long itemsets. In *Knowledge Discovery and Data Mining*, pages 165–174, 1999. URL citeseer.ist.psu.edu/meretakis99extending.html.

Rada Mihalcea y Paul Tarau. Textrank: Bringing order into texts. In *Conference on Empirical Methods in Natural Language Processing*, Barcelona, Spain, 2004. URL http://acl.ldc.upenn.edu/acl2004/emnlp/pdf/Mihalcea.pdf.

T. M. Mitchell. *Machine Learning*. McGraw Hill, 1997.

M. A. Mittermayer y G. Knolmayer. Text mining systems for market response to news: A survey. Working paper, 2006. URL http://www.ie.iwi.unibe.ch/publikationen/berichte/resource/WP-184.pdf.

Dunja Mladenić. Feature subset selection in text learning. In Claire Nédellec y Céline Rouveirol, editors, *Proceedings of ECML-98, 10th European Conference on Machine Learning*, pages 95–100, Chemnitz, DE, 1998. Springer Verlag, Heidelberg, DE. URL http://www-ai.ijs.si/DunjaMladenic/papers/PWW/pwwECML98.ps.gz. Published in the "Lecture Notes in Computer Science" series, number 1398.

Dunja Mladenić. Web browsing using machine learning on text data. pages 288–303, 2003.

Dunja Mladenić y Marko Grobelnik. Word sequences as features in text-learning. In *Proceedings of ERK-98, the Seventh Electrotechnical and Computer Science Conference*, pages 145–148, Ljubljana, SL, 1998.

Dunja Mladenić y Marko Grobelnik. Feature selection on hierarchy of web documents. *Decision Support Systems*, 35(1):45–87, 2003.

Dunja Mladenić, Janez Brank, Marko Grobelnik, y Natasa Milić-Frayling. Feature selection using linear classifier weights: interaction with classification models. In Kalervo Järvelin, James Allan, Peter Bruza, y Mark Sanderson, editors, *Proceedings of SIGIR-04, 27th ACM International Conference on Research and Development in Information Retrieval*, pages 234–241, Sheffield, UK, 2004. ACM Press, New York, US. URL http://doi.acm.org/10.1145/1008992.1009034.

Alistair Moffat y Justin Zobel. Exploring the similarity space. *SIGIR Forum*, 32(1), 1998.

Dan I. Moldovan, Marius Pasca, Sanda M. Harabagiu, y Mihai Surdeanu. Performance issues and error analysis in an open-domain question answering system. *ACM Trans. Inf. Syst.*, 21(2):133–154, 2003.

Katharina Morik, Peter Brockhausen, y Thorsten Joachims. Combining Statistical Learning with a Knowledge-Based Approach - A Case Study in Intensive Care Monitoring. In *ICML '99: Proceedings of the Sixteenth International Conference on Machine Learning*, pages 268–277, San Francisco, CA, USA, 1999. Morgan Kaufmann Publishers Inc. ISBN 1-55860-612-2. URL http://portal.acm.org/citation.cfm?id=657612.

Alessandro Moschitti y Roberto Basili. Complex linguistic features for text classification: A comprehensive study. In Sharon McDonald y John Tait, editors, *Proceedings of ECIR-04, 26th European Conference on Information Retrieval Research*, pages 181–196, Sunderland, UK, 2004. Springer Verlag, Heidelberg, DE. URL http://springerlink.metapress.com/openurl.asp?genre=article&issn=0302-9743&volume=2997&spage=181. Published in the "Lecture Notes in Computer Science" series, number 2997.

David Nadeau y Satoshi Sekine. A survey of named entity recognition and classification. *Linguisticae Investigationes*, 30(1):3–26, January 2007. URL http://www.ingentaconnect.com/content/jbp/li/2007/00000030/00000001/art00002. Publisher: John Benjamins Publishing Company.

Stuart J Nelson, Michael Schopen, Allan G Savage, Jacque-Lynne Schulman, y Natalie Arluk. The mesh translation maintenance system: structure, interface design, and implementation. *Medinfo*, 11(Pt 1):67–69, 2004.

Günter Neumann y Sven Schmeier. Combining shallow text processing and machine learning in real world applications. In *In Proceedings of the IJCAI-99 workshop on Machine Learning for Information Filtering*, pages 55–60, 1999.

Universitat d'Alacant
Universidad de Alicante

Kamal Nigam y Rayid Ghani. Analyzing the applicability and effectiveness of co-training. In Arvin Agah, Jamie Callan, y Elke Rundensteiner, editors, *Proceedings of CIKM-00, 9th ACM International Conference on Information and Knowledge Management*, pages 86–93, McLean, US, 2000. ACM Press, New York, US. URL http://www.cs.cmu.edu/~knigam/papers/cotrain-CIKM00.pdf.

Kamal Nigam, Andrew McCallum, y Tom Mitchell. Semi-supervised text classification using EM. pages 33–56. 2006.

Payal Patel, Mosin Hasan, y Bhavesh Tanawala. Distributed high performance web crawler. *IJITR*, 1(3):236–239, 2013.

Viktor Pekar, Michael Krkoska, y Steffen Staab. Feature weighting for co-occurrence-based classification of words. In *COLING '04: Proceedings of the 20th international conference on Computational Linguistics*, page 799, Morristown, NJ, USA, 2004. Association for Computational Linguistics. doi: http://dx.doi.org/10.3115/1220355.1220470.

C. Peters y Cornelis H. Koster. Uncertainty-based noise reduction and term selection in text categorization. In Fabio Crestani, Mark Girolami, y Cornelis J. Van Rijsbergen, editors, *Proceedings of ECIR-02, 24th European Colloquium on Information Retrieval Research*, pages 248–267, Glasgow, UK, 2002. Springer Verlag, Heidelberg, DE. URL http://link.springer.de/link/service/series/0558/papers/2291/22910248.pdf. Published in the "Lecture Notes in Computer Science" series, number 2291.

Jan Pomikálek, Pavel Rychlý, y Adam Kilgarriff. Scaling to billion-plus word corpora, 2009.

Martin F. Porter. An algorithm for suffix stripping. *Program*, 14(3):130–137, 1980.

J. R. Quinlan. Induction of decision trees. *Machine Learning*, 1(1):81–106, 1986. ISSN 0885-6125. doi: http://dx.doi.org/10.1023/A:1022643204877.

Muhammad Rafi y Mohammad Shahid Shaikh. A comparison of svm and rvm for document classification. *arXiv preprint arXiv:1301.2785*, 2013.

Yury Yineth Niño Roa. *Minería de Textos Médicos Web, un Enfoque Hacia la Clasificación en Categorías*. Seminario de Investigación I., 2008.

Stephen E. Robertson y Karen Sparck Jones. Relevance weighting of search terms. pages 143–160, 1988.

J. Rocchio. *Relevance Feedback in Information Retrieval*, pages 313–323. 1971. URL http://scholar.google.com/scholar?hl=en&lr=&client=firefox-a&q=relevance+feedback+in+information+retrieval&btnG=Search.

Tony Rose, Mark Stevenson, y Miles Whitehead. The Reuters Corpus Volume 1 – from yesterday's news to tomorrow's language resources. In *Proceedings of LREC-02, 3rd International Conference on Language Resources and Evaluation*, pages 827–832, Las Palmas, ES, 2002.

Dan Roth. Learning to resolve natural language ambiguities: a unified approach. In *Proceedings of AAAI-98, 15th Conference of the American Association for Artificial Intelligence*, pages 806–813, Madison, US, 1998. AAAI Press, Menlo Park, US. URL http://l2r.cs.uiuc.edu/~danr/Papers/aaai98.ps.gz.

Zhao Ru, Jun Guo, y Weiran Xu. *Improving Expertise Recommender Systems by Odds Ratio*, pages 1–9. 2008.

Patrick Ruch. Automatic assignment of biomedical categories: toward a generic approach. *Bioinformatics, volume 22 no. 6 2006*, pages 658–664, 2005.

Miguel Ruiz y Padmini Srinivasan. Hierarchical text classification using neural networks. *Information Retrieval*, 5(1):87–118, 2002. URL http://www.wkap.nl/article.pdf?383232.

Fernando Ruiz-Rico y Jose-Luis Vicedo. Newpar: An optimized feature selection and weighting schema for category ranking. *Frontiers in Artificial Intelligence and Applications*, 142:269, 2006.

Fernando Ruiz-Rico, Jose-Luis Vicedo, y María-Consuelo Rubio-Sánchez. Newpar: an automatic feature selection and weighting schema for category ranking. In *Proceedings of the 2006 ACM symposium on Document engineering*, pages 128–137. ACM, 2006.

Fernando Ruiz-Rico, Jose-Luis Vicedo, y María-Consuelo Rubio-Sánchez. Multilingual assistant for medical diagnosing and drug prescription based on category ranking. In *22nd International Conference on on Computational Linguistics: Demonstration Papers*, pages 169–172. Association for Computational Linguistics, 2008a.

Fernando Ruiz-Rico, Jose-Luis Vicedo, y María-Consuelo Rubio-Sánchez. Medline abstracts classification, average-based discrimination for noun phrases selection and weighting applied to categorization of medline abstracts. In Lus Azevedo y Ana Rita Londral, editors, *HEALTHINF (1)*, pages 94–101. INSTICC - Institute for Systems and Technologies of Information, Control and Communication, 2008b. ISBN 978-989-8111-16-6. URL http://dblp.uni-trier.de/db/conf/biostec/healthinf2008-1.html#Ruiz-RicoGR08.

Fernando Ruiz-Rico, Jose-Luis Vicedo, y María-Consuelo Rubio-Sánchez. Medical diagnosis assistant based on category ranking. In Lus Azevedo y Ana Rita Londral, editors, *HEALTHINF (1)*, pages 239–241. INSTICC - Institute for Systems and Technologies of Information, Control and Communication, 2008c. ISBN 978-989-8111-16-6. URL http://dblp.uni-trier.de/db/conf/biostec/healthinf2008-1.html#Ruiz-RicoGR08a.

Fernando Ruiz-Rico, Jose-Luis Vicedo, y María-Consuelo Rubio-Sánchez. Medline abstracts classification based on noun phrases extraction. *Biomedical Engineering Systems and Technologies*, pages 507–519, 2009.

Fernando Ruiz-Rico, David Tomás, Jose-Luis Vicedo, y María-Consuelo Rubio-Sánchez. A multilingual and multiplatform application for medicinal plants prescription from medical symptoms. In *Proceedings of SIGIR-13, 36th ACM International Conference on Research and Development in Information Retrieval*. ACM Press, New York, US, 2013a.

Fernando Ruiz-Rico, David Tomás, Jose-Luis Vicedo, y María-Consuelo Rubio-Sánchez. A portable multilingual medical directory by automatic categorization of wikipedia articles. In *Proceedings of SIGIR-13, 36th ACM International Conference on Research and Development in Information Retrieval*. ACM Press, New York, US, 2013b.

M. Sahami. *Using Machine Learning to Improve Information Access*. PhD thesis, Department of Computer Science. Standford University, 1998a.

Mehran Sahami, editor. *A Bayesian approach to Itering junk e-mail*, Madison, US, 1998b. Available as Technical Report WS-98-05.

G. Salton. *Automatic Text Processing: The Transformation, Analysis, and Retrieval of Information by Computer*. Addison-Wesley, 1989.

G. Salton, A. Wong, y C.S. Yang. Term-weighting approaches in automatic text retrieval. *Information Processing and Management*, 5(24):513–523, 1988.

Robert E. Schapire y Yoram Singer. BoosTexter: a boosting-based system for text categorization. *Machine Learning*, 39(2/3):135–168, 2000. URL http://www.research.att.com/~schapire/papers/SchapireSi98b.ps.Z.

Robert E. Schapire, Yoram Singer, y Amit Singhal. Boosting and rocchio applied to text filtering. In W. Bruce Croft, Alistair Moffat, Cornelis J. Van Rijsbergen, Ross Wilkinson, y Justin Zobel, editors, *Proceedings of SIGIR-98, 21st ACM International Conference on Research and Development in Information Retrieval*, pages 215–223, Melbourne, AU, 1998. ACM Press, New York, US. URL http://www.research.att.com/~schapire/cgi-bin/uncompress-papers/SchapireSiSi98.ps.

J. Schürmann. *Pattern Classification: A Unified View of Statistical and Neural Approaches*. John Wiley & Sons, New York, 1996.

Hinrich Schütze, David A. Hull, y Jan O. Pedersen. A comparison of classifiers and document representations for the routing problem. In Edward A. Fox, Peter Ingwersen, y Raya Fidel, editors, *Proceedings of SIGIR-95, 18th ACM International Conference on Research and Development in Information Retrieval*, pages 229–237, Seattle, US, 1995. ACM Press, New York, US. URL ftp://parcftp.xerox.com/pub/qca/papers/sigir95.ps.gz.

Sam Scott. Feature engineering for a symbolic approach to text classification. Master's thesis, Computer Science Department, University of Ottawa, Ottawa, CA, 1998. URL http://ai.iit.nrc.ca/II_public/Classification/thesis.pdf.

Sam Scott y Stan Matwin. Feature engineering for text classification. In Ivan Bratko y Saso Dzeroski, editors, *Proceedings of ICML-99, 16th International Conference on Machine Learning*, pages 379–388, Bled, SL, 1999. Morgan Kaufmann Publishers, San Francisco, US.

Fabrizio Sebastiani. A tutorial on automated text categorisation. In Analia Amandi y Ricardo Zunino, editors, *Proceedings of ASAI-99, 1st Argentinian Symposium on Artificial Intelligence*, pages 7–35, Buenos Aires, AR, 1999. URL http://www.math.unipd.it/~fabseb60/Publications/ASAI99.pdf. An extended version appears as Sebastiani [2002].

Fabrizio Sebastiani. Machine learning in automated text categorization. *ACM Computing Surveys*, 34(1):1–47, 2002. URL http://www.math.unipd.it/~fabseb60/Publications/ACMCS02.pdf.

Amit Singhal, Chris Buckley, y Mandar Mitra. Pivoted document length normalization. *Department of Computer Science, Cornell University, Ithaca, NY 14853*, 1996.

Alan Smeaton. Information retrieval: Still butting heads with natural language processing ? In *Information Technology, M.T Pazienza ed., Springer-Verlag Lecture Notes in Computer Scienc, number 1299*, pages 115–138. Springer-Verlag, 1997.

R. L. Solomon y D. H. Howes. Word frequency, personal values, and visual duration thresholds. *Psychol Rev*, 58(4):256–70, 1951. ISSN 0033-295X. URL http://www.biomedsearch.com/nih/Word-frequency-personal-values-visual/14864763.html.

Pascal Soucy y Guy W. Mineau. Beyond tfidf weighting for text categorization in the vector space model. In *IJCAI'05: Proceedings of the 19th international joint conference on Artificial intelligence*, pages 1130–1135, San Francisco, CA, USA, 2005. Morgan Kaufmann Publishers Inc.

K. Sparck-Jones. Index term weighting, 1973.

H Sundblad. *Question Classification in Question Answering Systems*. Linköping studies in science and technology: Thesis. Department of Computer and Information Science, Linköpings universitet, 2007. ISBN 9789185831555. URL http://books.google.es/books?id=vf1ENQAACAAJ.

Chade-Meng Tan, Yuan-Fang Wang, y Chan-Do Lee. The use of bigrams to enhance text categorization. *Information Processing and Management*, 38(4):529–546, 2002. URL http://www.serve.com/cmtan/Meng/ig_m.pdf.

Daniel R. Tauritz, Joost N. Kok, y Ida G. Sprinkhuizen-Kuyper. Adaptive information filtering using evolutionary computation. *Information Sciences*, 122(2/4):121–140, 2000. URL http://www.elsevier.nl/gej-ng/10/23/143/56/27/27/article.pdf.

David Tax y Robert Duin. Combining one-class classifiers. *Multiple Classifier Systems*, pages 299–308, 2001.

Roman Tesar, Vaclav Strnad, Karel Jezek, y Massimo Poesio. Extending the single words-based document model: a comparison of bigrams and 2-itemsets. In *DocEng '06: Proceedings of the 2006 ACM symposium on Document engineering*, pages 138–146, New York, NY, USA, 2006. ISBN 1-59593-515-0. doi: http://doi.acm.org/10.1145/1166160.1166197.

Bernd Teufel y Stephanie Schmidt. Full text retrieval based on syntactic similarities. *Information Systems*, 13(1):65–70, 1988.

Jörg Tiedemann. News from OPUS - A collection of multilingual parallel corpora with tools and interfaces. In N. Nicolov, K. Bontcheva, G. Angelova, y R. Mitkov, editors, *Recent Advances in Natural Language Processing*, volume V, pages 237–248. John Benjamins, Amsterdam/Philadelphia, Borovets, Bulgaria, 2009. ISBN 978 90 272 4825 1.

David Tomás. *Sistemas de clasificación de preguntas basados en corpus para la búsqueda de respuestas*. PhD thesis, Universidad de Alicante, 2009.

David Tomás, José L. Vicedo, y Armando Suárez. Una aproximación multilingüe a la clasificación de preguntas basada en aprendizaje automático. 2005. URL http://www.sepln.org/revistaSEPLN/revista/35/48.pdf.

K. Tumer y J. Ghosh. Error correlation and error reduction in ensemble classifiers. *Connection Science*, 8(3/4):385–404, 1996.

Universidad Valencia. Filtro antipropaganda (anti-spam) en el servicio de correo electrónico. Technical report, Universidad de Valencia (España). Servicio de Informática, 2012. Accesible desde http://www.uv.es/siuv/cas/zinternet/filtres.htm.

Alberto Téllez Valero. *Extracción de Información con Algoritmos de Clasificación. Tesis de Maestro en Ciencias Computacionales*. Instituto Nacional de Astrofísica, Óptica y Electrónica., 2005.

Cornelis J. Van Rijsbergen. *Information Retrieval, 2nd edition*. Dept. of Computer Science, University of Glasgow, 1979.

Andreas S. Weigend, Erik D. Wiener, y Jan O. Pedersen. Exploiting hierarchy in text categorization. *Information Retrieval*, 1(3):193–216, 1999. URL http://www.stern.nyu.edu/~aweigend/Research/Papers/TextCategorization/hierarchy.ps.

Stefan Wermter y Chihli Hung. Selforganizing classification on the reuters news corpus. In *Proceedings of COLING-02, the 19th International Conference on Computational Linguistics*, Taipei, TW, 2002. URL http://www.his.sunderland.ac.uk/ps/coling-232.pdf.

Bernard Widrow, Michael, y A. Lehr. 30 years of adaptive neural networks: Perceptron, madaline and backpropagation. pages 1415–1442. IEEE press, 1990.

Erik D. Wiener, Jan O. Pedersen, y Andreas S. Weigend. A neural network approach to topic spotting. In *Proceedings of SDAIR-95, 4th Annual Symposium on Document Analysis and Information Retrieval*, pages 317–332, Las Vegas, US, 1995. URL http://www.stern.nyu.edu/~aweigend/Research/Papers/TextCategorization/Wiener.Pedersen.Weigend_SDAIR95.ps.

Universitat d'Alacant
Universidad de Alicante

Peter Willett. Similarity coefficients and weighting functions for automatic document classification: An empirical comparison, 1983.

Hongzhi Xu y Chunping Li. A novel term weighting scheme for automated text categorization. In *ISDA '07: Proceedings of the Seventh International Conference on Intelligent Systems Design and Applications*, pages 759–764, Washington, DC, USA, 2007. IEEE Computer Society. ISBN 0-7695-2976-3.

Yiming Yang. An evaluation of statistical approaches to medline indexing. In James J. Cimino, editor, *Proceedings of AMIA-96, Fall Symposium of the American Medical Informatics Association*, pages 358–362, Washington, US, 1996. Hanley and Belfus. URL http://www.cs.cmu.edu/afs/cs/user/yiming/www/courses/bibliography/papers/scamc96.ps.

Yiming Yang. An evaluation of statistical approaches to text categorization. *Information Retrieval*, 1(1/2):69–90, 1999. URL http://www.cs.cmu.edu/~yiming/papers.yy/irj99.ps.

Yiming Yang y Christopher G. Chute. An example-based mapping method for text categorization and retrieval. *ACM Transactions on Information Systems*, 12(3):252–277, 1994. URL http://www.acm.org/pubs/articles/journals/tois/1994-12-3/p252-yang/p252-yang.pdf.

Yiming Yang y Xin Liu. A re-examination of text categorization methods. In Marti A. Hearst, Fredric Gey, y Richard Tong, editors, *Proceedings of SIGIR-99, 22nd ACM International Conference on Research and Development in Information Retrieval*, pages 42–49, Berkeley, US, 1999. ACM Press, New York, US. URL http://www.cs.cmu.edu/~yiming/papers.yy/sigir99.ps.

Yiming Yang y Jan O. Pedersen. A comparative study on feature selection in text categorization. In Douglas H. Fisher, editor, *Proceedings of ICML-97, 14th International Conference on Machine Learning*, pages 412–420, Nashville, US, 1997. Morgan Kaufmann Publishers, San Francisco, US. URL http://www.cs.cmu.edu/~yiming/papers.yy/ml97.ps.

Yiming Yang y John W. Wilbur. Using corpus statistics to remove redundant words in text categorization. *Journal of the American Society for Information Science*, 47(5):357–369, 1996. URL http://www3.interscience.wiley.com/cgi-bin/fulltext?ID=57757&PLACEBO=IE.pdf.

Yiming Yang, Seán Slattery, y Rayid Ghani. A study of approaches to hypertext categorization. *Journal of Intelligent Information Systems*, 18(2/3):219–241, 2002. URL http://www.wkap.nl/article.pdf?391248. Special Issue on Automated Text Categorization.

Yiming Yang, Jian Zhang, y Bryan Kisiel. A scalability analysis of classifiers in text categorization. In Jamie Callan, Gordon Cormack, Charles Clarke, David Hawking, y Alan Smeaton, editors, *Proceedings of SIGIR-03, 26th ACM International Conference on Research and Development in Information Retrieval*, pages 96–103, Toronto, CA, 2003. ACM Press, New York, US. URL http://doi.acm.org/10.1145/860435.860455.

Universitat d'Alacant
Universidad de Alicante

Y. Jianfeng Z. Yang, Z. Lijun y L. Zhanhuai. Using association features to enhance the performance of naive bayes text classifier. *Fifth International Conference on Computational Intelligence and Multimedia Applications, ICCIMA '03*, pages 336–341, 2003.

Zhaohui Zheng y Rohini Srihari. Optimally Combining Positive and Negative Features for Text Categorization. In *Workshop for Learning from Imbalanced Datasets II, Proceedings of the ICML*, Washington, DC, 2003.

Zhaohui Zheng, Xiaoyun Wu, y Rohini Srihari. Feature selection for text categorization on imbalanced data. *SIGKDD Explorations*, 6(1):80–89, 2004. URL http://doi.acm.org/10.1145/1007730.1007741.

Dong-Qing Yang-Ming Zhang Li-Yu Li Zhi-Hong Deng, Shi-Wei Tang y Kun-Qing Xie. *A Comparative Study on Feature Weight in Text Categorization*, pages 588–597. 2004.

Xiaojin Zhu. Semi-supervised learning literature survey. Technical Report 1530, Computer Sciences, University of Wisconsin-Madison, 2005.

George K. Zipf. *Human Behavior and the Principle of Least Effort*. Addison-Wesley (Reading MA), 1949.

Guowei Zu, Wataru Ohyama, Tetsushi Wakabayashi, y Fumitaka Kimura. Accuracy improvement of automatic text classification based on feature transformation. In *Proceedings of DOCENG-03, ACM Symposium on Document engineering*, pages 118–120, Grenoble, FR, 2003. ACM Press, New York, US. URL http://doi.acm.org/10.1145/958220.958242.

Universitat d'Alacant
Universidad de Alicante