Extracción de conocimiento de microarrays y literatura biomédica para el estudio de la regulación genética.

Memoria para la obtención del Título de Doctor

Carlos Cano Gutiérrez

Director: Armando Blanco Morón

Universidad de Granada

Dpto. Ciencias de la Computación e Inteligencia Artificial
Universidad de Granada

Febrero 2010

Editor: Editorial de la Universidad de Granada
Autor: Carlos Cano Gutiérrez
D.L.: GR 2322-2010
ISBN: 978-84-693-1323-7

Towards the identification of gene regulatory mechanisms: knowledge extraction from microarrays and the biomedical literature.

PhD Dissertation

Carlos Cano Gutiérrez

Supervisor: Armando Blanco Morón

Universidad de Granada

Dpto. Ciencias de la Computación e Inteligencia Artificial
Universidad de Granada

February 2010

La memoria “Extracción de conocimiento de microarrays y literatura biomédica para el estudio de la regulación genética”, que presenta D. Carlos Cano Gutiérrez para optar al grado de Doctor en Informática, ha sido realizada en el Departamento de Ciencias de la Computación e Inteligencia Artificial de la Universidad de Granada bajo la supervisión del Profesor Dr. D. Armando Blanco Morón, profesor titular de universidad.

Granada, Enero de 2010.

Armando Blanco Morón

Carlos Cano Gutiérrez

Índice

Introducción **xvii**

Antecedentes xvii

Objetivos xviii

Estructura de la memoria xix

I Preliminares **1**

1 Preliminares **3**

1.1 Introducción a la tecnología de microarrays de expresión. 3

1.1.1 Microarrays. Fundamentos biológicos. 4

1.1.2 Tecnología de microarrays de expresión genética. 8

1.1.3 Análisis de microarrays de expresión. 11

1.1.4 Aplicaciones de los microarrays. 14

1.1.5 Limitaciones de los microarrays de expresión genética. 16

1.2 Aprendizaje automático y bioinformática 20

1.2.1 Clustering 24

1.2.2 Clustering difuso. 27

1.2.3 Biclustering. 30

1.2.4 Clasificación. 33

1.3 Sistemas de extracción de información de textos biomédicos. 40

1.3.1 Análisis automático de la literatura biomédica. 40

1.3.2 Análisis automático de historias clínicas. 59

1.4 Sistemas de Anotación de textos biomédicos 62

1.4.1 Anotación social y colaborativa en las ciencias de la vida 62

1.4.2 La Web Semántica en las ciencias de la vida. 64

II Análisis de microarrays mediante Clustering y Biclustering 67

2 Clustering y biclustering para identificar patrones de máxima varianza en matrices de expresión genética 69
2.1 Motivación y objetivos 69
2.2 Aplicación del clustering al análisis de microarrays 71
2.3 Aplicación del biclustering al análisis de microarrays 74
2.4 Algoritmo de partida: Gene Shaving. 79
2.4.1 Objetivos. 79
2.4.2 Descripción del algoritmo. 79
2.5 Limitaciones del algoritmo Gene Shaving. 83
2.6 Aplicación de algoritmos evolutivos a la obtención de clusters de máxima varianza 85
2.6.1 Clustering utilizando Algoritmos Genéticos: GA-Clustering . . . 86
2.6.2 Clustering utilizando EDAs: EDA-Clustering 87
2.7 Biclustering para identificar patrones de máxima varianza 89
2.7.1 Biclustering utilizando Componentes Principales: Gene&Sample Shaving. 89
2.7.2 Biclustering utilizando EDAs: EDA-Biclustering. 91
2.8 Experimentos y Análisis de Resultados. 93
2.8.1 Algoritmos de Clustering. 95
2.8.2 Algoritmos de biclustering. 97
2.9 Conclusiones. 101

3 Biclustering espectral posibilístico en matrices de expresión genética 105
3.1 Motivación y objetivos. 105
3.2 Aplicación del clustering difuso al análisis de microarrays 107
3.3 Enfoque de partida: biclustering espectral. 108
3.3.1 Algoritmo de Dhillon. 113
3.4 Limitaciones de los algoritmos espectrales de biclustering. 114
3.5 Biclustering espectral posibilístico. 114
3.5.1 Tecnología difusa para el clustering de microarrays. 115
3.5.2 Varias matrices de partición. 119
3.5.3 Agrupamiento de filas y columnas independientemente. 120
3.5.4 Biclusters crisp a partir de biclusters posibilísticos. 120
3.5.5 Limitación de solapamiento entre biclusters. 122
3.5.6 Inversión de la matriz de expresión. 122
3.5.7 Pseudocódigo del algoritmo PSB. 122
3.6 Experimentos y análisis de resultados. 123
3.7 Conclusiones. 132

III Anotación manual y sistemas automáticos para la extracción de conocimiento de textos biomédicos. 135

4 Sistemas de anotación colaborativa de textos biomédicos 137
4.1 Motivación y objetivos . . . 137
4.2 Corpora en biomedicina . . . 141
4.3 Herramientas de anotación de textos . . . 146
4.4 BioNotate: una herramienta de anotación colaborativa para textos biomédicos . . . 149
4.4.1 Esquema y proceso de anotación . . . 149
4.4.2 Sistema de Anotación . . . 154
4.5 Caso de estudio: corpus piloto sobre autismo . . . 159
4.5.1 Motivación y Objetivos. . . . 159
4.5.2 Fuentes de datos y métodos para la creación del corpus. . . . 162
4.5.3 Resultados sobre el corpus piloto. . . . 164
4.6 Extensiones y aplicaciones de BioNotate. . . . 168
4.6.1 Flexibilización del esquema y proceso de anotación. . . . 168
4.6.2 AutismNotate. . . . 170
4.6.3 BioNotate y la Web Semántica. . . . 174
4.7 Conclusiones. . . . 180

5 Identificación automática de diagnósticos en historias clínicas. Caso de estudio en obesidad. 183
5.1 Motivación y objetivos. . . . 183
5.2 Trabajo previo en identificación automática de diagnosis en historias clínicas. . . . 185
5.3 Descripción del corpus de historiales clínicos. . . . 186
5.4 Clasificación basada en Regresión Logística para Historias Clínicas. . . . 187
5.4.1 Regresión Logística. . . . 187
5.4.2 Conjuntos de características. . . . 189
5.4.3 Arquitectura del sistema. . . . 191
5.5 Experimentación y Resultados. . . . 193
5.5.1 Evaluación comparativa de la Regresión Logística para clasificación de textos. . . . 193
5.5.2 Evaluación comparativa de distintas representaciones de características. . . . 195
5.5.3 Análisis detallado de las distintas estrategias . . . 196
5.5.4 Intentos fallidos para mejorar la bondad de resultados. . . . 199
5.6 Conclusiones. . . . 203

IV Conclusiones **207**

6 Conclusiones y trabajo futuro. **209**

6.1 Conclusiones 209

6.2 Trabajo Futuro 212

7 Conclusions and future work **217**

7.1 Conclusions 217

7.2 Future work 219

V Publicaciones **223**

8 Trabajos publicados **225**

Bibliografía **231**

Índice de figuras

1.1 Estructura de la doble hélice de ADN 5
1.2 Visión esquemática de la síntesis de proteínas. 6
1.3 Visión tradicional del dogma central de la biología molecular. 7
1.4 Pasos de un experimento de análisis de microarrays de expresión. 9
1.5 Pasos para la hibridación en un microarray de expresión de ADNc. 10
1.6 Microarray de expresión de dos colores tras ser excitado con láser 11
1.7 Matriz de expresión genética 12
1.8 El papel de los intrones y exones en la transcripción 18
1.9 Taxonomía de métodos de aprendizaje automático 22
1.10 Tipos de bicluster según el criterio de correlación de valores 31
1.11 Tipos de bicluster según estructura y restricciones de solapamiento 32
1.12 Definición de *true positive*, *true negative*, *false positive* y *false negative* en un problema de clasificación 38
1.13 Evolución del volumen de artículos en MEDLINE. 41

2.1 Ejemplo de la aplicación de clustering jerárquico para agrupar muestras en una matriz de expresión. 73
2.2 Ejemplo ilustrativo de la utilidad del biclustering 75
2.3 Ejemplo de aplicación de Gene Shaving 80
2.4 Esquema de la resolución del problema de encontrar k genes que maximicen la varianza para las muestras de A. 81
2.5 Dos esquemas de implementación de la selección de genes utilizando Algoritmos Evolutivos. 85
2.6 Esquema de un Algoritmo Genético Generacional 86
2.7 Esquema general de un algoritmo EDA. 88
2.8 Esquema del algoritmo Gene&Sample Shaving. 91
2.9 Algoritmo general EDA. 92
2.10 Diagrama de dispersión del GAP y tamaño de los clusters obtenidos con los distintos algoritmos de clustering en el *dataset* de la levadura 96

2.11 Perfiles de expresión para algunos clusters biológicamente significativos encontrados con EDA-Clustering 97
2.12 Perfiles de expresión genética para clusters significativamente asociados a *DNA replication* obtenidos con Gene-Shaving, GA-Clustering, *multiple-step* EDA-Clustering y *single-step* EDA-Clustering 98
2.13 Perfiles de expresión genética para clusters significativamente asociados a *DNA unwinding* obtenidos con Gene-Shaving y *single-step* EDA-Clustering 99
2.14 Perfiles de expresión genética para algunos biclusters obtenidos con Gene&Sample Shaving y EDA Biclustering 100
2.15 Perfiles de expresión para los genes contenidos en un bicluster obtenido con EDA Biclustering sobre los datos de linfoma humano 102

3.1 Ejemplo representación de una matriz de expresión mediante un grafo bipartito 109
3.2 Representación gráfica del algoritmo PSB 121
3.3 Evolución del número y calidad de los biclusters obtenidos en el dataset de la levadura utilizando de 1 a 15 eigenvectores. 127
3.4 *Correspondence plots* para biclusters generados con PSB, FLOC, Cheng&Church, PLAID y biclusters aleatorios en los datos de la levadura. 128
3.5 *Correspondence plots* para biclusters obtenidos reemplazando al azar el 10,20,30,40 y 50 % de los genes de cada bicluster obtenido por PSB en el dataset de la levadura. 129
3.6 Evolución del número y calidad de los biclusters obtenidos en el dataset de linfomas utilizando de 1 a 20 eigenvectores. 130
3.7 *Correspondence plots* para biclusters obtenidos con PSB, FLOC, Cheng&Church, PLAID y biclusters aleatorios sobre dataset de linfomas. 131
3.8 *Correspondence plots* para la significación de tipos de muestras en los biclusters obtenidos por PSB, FLOC, Cheng&Church, PLAID y biclusters aleatorios para el dataset de linfomas 133

4.1 Extracción de redes de interacción entre entidades biomédicas por medio del análisis de la literatura. 139
4.2 Captura de pantalla de la aplicación LabelMe 148
4.3 Esquema del proceso de anotación de un snippet 157
4.4 Esquema de la arquitectura de BioNotate 159
4.5 Extracto de un snippet en formato XML. 160
4.6 Captura de pantalla del interfaz de anotación de BioNotate 160

4.7 Flujo de información de entrada y salida de BioNotate. 161
4.8 Flujo de entrada y salida de la nueva versión de BioNotate 169
4.9 Página principal del proyecto AutismNotate 172
4.10 Interfaz de anotación de AutismNotate construida sobre BioNotate 173
4.11 Integración de anotaciones manuales y automáticas en BioNotate y publicación de las anotaciones utilizando tecnologías de la Web Semántica . . 175
4.12 Captura de pantalla del sistema de anotación BioNotate que muestra la posibilidad de normalizar las entidades marcadas contra ontologías y recursos terminológicos. 178

5.1 Arquitectura del sistema clasificador de historiales clínicos. 192
5.2 Evolución de la tasa de acierto promedio para la clasificación intuitiva con léxicos reducidos. 200
5.3 Evolución de la tasa de acierto promedio para la clasificación textual con léxicos reducidos. 201
5.4 *Heatmaps* con distintas medidas de distancia entre cada pareja de enfermedades 202

6.1 Esquema del ciclo de aprendizaje con *active learning* 214

Índice de tablas

1.1 Principales plataformas software para análisis de datos de microarrays. . 15
1.2 Principales bases de datos utilizadas en biología molecular. 21
1.3 Extractos de textos biomédicos. 43
1.4 Ejemplos de NEs de conceptos biomédicos. 44
1.5 Evaluaciones competitivas y resutados de los mejores sistemas de NER . . 48
1.6 Extractos de historiales clínicos reales. 60

2.1 *Dataset* de la levadura. Valores medios y desviaciones típicas del GAP y tamaño para 100 clusters. 95
2.2 *Dataset* de la levadura. Valores medios y desviaciones típicas del GAP y tamaño para los biclusters obtenidos. 99
2.3 *Dataset* de linfomas. Valores medios y desviaciones típicas del GAP y tamaño de los agrupamientos obtenidos. 99

3.1 Comparativa de resultados de algoritmos de biclustering sobre datos sintéticos con un bicluster. 125
3.2 Comparativa de resultados de algoritmos de biclustering sobre datos sintéticos con tres biclusters. 126
3.3 Resultados de PSB y comparativa con otros métodos para el dataset de la levadura. 127
3.4 Procesos biológicos de GO más significativos de los biclusters obtenidos por PSB para el dataset de la levadura. 129
3.5 Resultados de PSB y comparativa con otros métodos en el dataset de linfomas. 131

4.1 Resumen de las características de los distintos corpora con anotaciones de entidades, sintácticas y relaciones proteína-proteína. 143
4.2 Características de los corpora con anotaciones para relaciones Proteína-Proteína. 145
4.3 Extractos de textos biomédicos reales con menciones a genes, proteínas y enfermedades. 150

4.4 Ejemplos de snippets anotados 154
4.5 Acuerdo entre anotadores tras la anotación del corpus piloto 167

5.1 Representación de un modelo de regresión logística entrenado 194
5.2 Comparativa de acierto promedio por enfermedad para la clasificación textual 195
5.3 Micro y Macro *precision*, *recall* y *f-measure* del clasificador de regresión logística 196
5.4 Efecto de la adición de categorías de fármacos en la tasa de acierto promedio por enfermedad 198
5.5 Efecto del preprocesamiento de frases con negación en la tasa de acierto por enfermedad 198
5.6 Acierto promedio para clasificación textual con las respuestas añadidas para las otras enfermedades y sin ellas 203
5.7 Acierto promedio para clasificación intuitiva con las respuestas añadidas para las otras enfermedades y sin ellas 204

Resumen

Con la disponibilidad de los genomas completos de un creciente número de organismos, la bioinformática se ha erigido como un pilar imprescindible en la investigación y desarrollo de las ciencias biomédicas, jugando un papel esencial tanto en el análisis de datos genómicos y proteómicos generados por las tecnologías de altas prestaciones, como en la organización y almacenamiento de información derivada de estas tecnologías.

Esta memoria presenta varias aportaciones al estudio de los mecanismos de regulación celulares y la base genética de las enfermedades, mediante el análisis de datos producidos por tecnologías de microarrays de expresión y la información contenida en textos biomédicos y clínicos.

Respecto al análisis de datos de expresión genética, se proponen varios algoritmos de clustering y biclustering no-exclusivos para la identificación de grupos de genes que exhiban patrones de expresión similares. Los métodos propuestos utilizan distintos criterios de optimización e implementan diversos paradigmas computacionales, desde el análisis de componentes principales, hasta algoritmos evolutivos o la lógica difusa.

Respecto al análisis de textos biomédicos, en esta memoria se presentan dos contribuciones: una herramienta web de código abierto para la anotación colaborativa de textos biomédicos con la asistencia de herramientas de text-mining, y un clasificador basado en regresión logística, para la identificación automática de diagnósticos de interés en el texto de historias clínicas.

Abstract

With an increasing number of sequenced genomes, bioinformatics has become a fundamental part of biomedical research, playing a key role in the analysis of genomic and proteomic data produced by high-throughput technologies and in the organization and storage of information derived from these technologies.

This dissertation provides several novel contributions to the decomposition and ultimate interpretation of the genetic basis of disease based on the analysis of high-throughput gene expression data and the biomedical literature.

We accomplish the analysis of gene expression data by using several non-exclusive clustering and biclustering algorithms for the identification of potential functional modules. The proposed methods explore different optimization criteria and make use of several paradigms, ranging from principal component analysis to evolutionary algorithms and fuzzy technology.

With respect to the analysis of biomedical text, this dissertation presents two important contributions to the field: an open-source web-based annotation tool that allows the biomedical community to carry out collaborative curation efforts assisted by NLP-based tools; and a methodology for rapidly tailoring a logistic regression classifier to the automatic identification of diagnoses in clinical discharge summaries.

Agradecimientos

Me gustaría aprovechar estas líneas para expresar mi agradecimiento a las personas que, de una forma u otra, me han ayudado durante estos años.

En primer lugar, quiero expresar mi gratitud a mi director de tesis, Armando Blanco, por compartir conmigo su experiencia, tanto en lo profesional como en lo personal, y convertirse en un gran tutor, y en un mejor amigo.

También me gustaría agradecer a los miembros del Departamento de Ciencias de la Computación e Inteligencia Artificial y a los becarios de la sala 16, por compartir desayunos, fútbol y risas. Especialmente, me gustaría agradecer a mis compañeros de grupo Marta, Javi, Fernando, Luis y Alberto, por compartir tantos buenos ratos y estar siempre dispuestos a echar una mano.

Tengo que agradecer el apoyo económico prestado por el Ministerio de Educación y Ciencia, que me ha financiado durante estos años a través del programa de Formación de Profesorado Universitario. El trabajo de esta memoria también ha sido financiado, en parte, por los proyectos TIC 2003-09331-C02-01, Madrid; P05-TIC-640, Sevilla; TIN-2006-13177, Madrid y P08-TIC-04299, Sevilla.

Parte de esta tesis ha sido desarrollada durante mi estancia en la Universidad de Harvard, EEUU. Quiero agradecer especialmente a Leon Peshkin por su tutela, atención y apoyo durante aquel año, y por estar siempre dispuesto a compartir Ciencia conmigo en cualquier café de Cambridge. También a todos los compañeros del Center for Biomedical Informatics, especialmente a Dennis P. Wall y Peter Tonellato, que además han tenido la gentileza de emitir sendos informes apoyando la mención de esta tesis como Doctorado Internacional de la Universidad de Granada. Y como no, a todos mis buenos amigos de Boston, por abrirme sus casas y hacerme sentir como en la mía.

Me gustaría agradecer, de corazón, a mis amigos, los que siempre estáis ahí, por vuestro apoyo y afecto.

Y especialmente, quiero dedicar esta tesis a mi familia. A mis abuelos Ana, May y Lolo, porque cuando os veo siempre estoy orgulloso de vosotros; a mi hermana, Blanca, y mis padres, María Belén y Carlos, por vuestro apoyo infatigable y por darme el cariño y la confianza para perseguir siempre mis metas.

Y a Natalia, porque después de tanta investigación, lo que seguro he descubierto, es lo feliz que soy contigo.

A vosotros va dedicada esta tesis.

Introducción

Antecedentes

Las dos últimas décadas han supuesto una revolución en la investigación biomédica, debida, en parte, a un crecimiento exponencial del volumen de información disponible, generada, tanto por estudios clínicos y farmacológicos, como por tecnologías de altas prestaciones (*high-throughput*) en investigaciones genómicas y proteómicas. El rápido progreso de la biomedicina y la biotecnología han motivado la aparición y rápido crecimiento de un nuevo campo: la bioinformática.

La bioinformática, definida como el desarrollo y aplicación de técnicas y herramientas computacionales para el almacenamiento, organización y análisis de información biomédica, es una nueva disciplina, nacida para cubrir la necesidad de manejar las ingentes cantidades de información procedentes, tanto de la secuenciación de macromoléculas como, por ejemplo, ADN, proteínas y glúcidos, como de las técnicas de análisis masivo del comportamiento de genes y proteínas. Se trata de un área de investigación multidisciplinar a medio camino entre la Biología y las Ciencias de la Computación, que persigue como objetivo descubrir conocimiento a partir de estas grandes masas de datos que ayude a clarificar la regulación de los procesos celulares y los fundamentos que rigen el funcionamiento de los organismos vivos. Este conocimiento podría tener un gran impacto en campos tan variados como salud humana, agricultura, medio ambiente, energía y biotecnología, entre otros.

Los microarrays de expresión constituyen uno de los últimos avances en biología molecular, permitiendo la monitorización de la expresión de decenas de miles de genes en paralelo. Esta tecnología genera una enorme cantidad de datos, de forma que el análisis e interpretación manual de los mismos sería una tarea lenta, costosa y altamente subjetiva. Distintas técnicas estadísticas son habitualmente empleadas para el análisis de microarrays. No obstante, dada la naturaleza y volumen de los datos, surge la necesidad de aplicar en este campo herramientas y técnicas automáticas que

soporten la extracción de conocimiento útil a partir de los mismos, y estas técnicas se engloban bajo la denominación de Aprendizaje Automático (*Machine Learning*) y Minería de Datos (*Data Mining*). De especial interés y extensión dentro de las técnicas de aprendizaje automático para el análisis de microarrays resultan las técnicas de agrupamiento (Clustering y Biclustering) para encontrar grupos funcionales de genes. Estas técnicas se basan en la premisa de que si los genes muestran comportamientos similares para distintas muestras y condiciones experimentales, tendrán funciones biológicas relacionadas.

Los avances en la tecnología de microarrays y otras tecnologías de producción masiva de datos, están incrementando vertiginosamente el volumen de datos disponibles y, por tanto, el número de descubrimientos biológicos deducidos del análisis de estos datos. Todos los resultados obtenidos se publican en la literatura especializada, con lo que el volumen de artículos científicos está experimentando un crecimiento exponencial en los últimos años [149]. La búsqueda de términos médicos (nombres de genes, proteínas, enfermedades, etc.) en motores especializados como PubMed [1] devuelve miles de resultados de relevancia. En este contexto, en el que el volumen de información disponible desborda la capacidad humana de procesamiento y asimilación de la misma, se hace imprescindible la utilización de herramientas de Minería de Datos (o más específicamente, de Minería de Textos o *text-mining*) para extraer el conocimiento de forma automática de la literatura biomédica. Para el estudio de la regulación genética resulta de especial interés la detección de entidades biológicas como genes, proteínas y enfermedades en los textos biomédicos y la identificación de interacciones y asociaciones entre estas entidades descritas en la literatura.

Objetivos

El presente proyecto de investigación tiene como objetivo el desarrollo de métodos para la extracción de conocimiento que permita a los expertos avanzar en la identificación de los mecanismos de regulación genética. Más concretamente, el objetivo general es el desarrollo de metodologías y herramientas para el análisis y extracción de conocimiento de dos fuentes de datos biológicas:

- Datos de expresión genética obtenidos a partir de microarrays.

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

- Textos biomédicos.

Respecto al análisis de datos de microarrays, desarrollamos distintos algoritmos de clustering y biclustering para la identificación de agrupamientos no exclusivos en matrices de expresión genética. Los métodos propuestos se dividen en dos categorías. El primer conjunto de métodos utiliza el Análisis de Componentes Principales y distintos tipos de Algoritmos Evolutivos para obtener grupos de genes con perfiles de expresión similares, que además presenten varianza máxima para las distintas muestras en estudio. Por otro lado, desarrollamos un nuevo algoritmo de biclustering basado en técnicas espectrales y tecnología difusa para la identificación de patrones altamente coherentes en datos de expresión genética.

Respecto al análisis de la literatura y textos biomédicos, describimos distintas aproximaciones orientadas a la extracción de conocimiento de los mismos.

Por una parte, describimos una herramienta para la anotación de entidades biológicas de interés (genes, proteínas, enfermedades, etc.) y relaciones entre las mismas, en extractos de textos biomédicos. La creación de *corpora* (conjuntos de textos anotados) en este ámbito resulta fundamental para el desarrollo y mejora de métodos de text-mining basados en Procesamiento de Lenguaje Natural, para identificar este tipo de entidades y relaciones de forma automática. En particular, la herramienta desarrollada permite llevar a cabo esfuerzos distribuidos de anotación que combinan la anotación manual con sistemas automáticos de text-mining.

Por otro lado, presentamos un clasificador basado en Regresión Logística para la identificación automática de diagnósticos a partir del texto de historias clínicas.

Estructura de la memoria

Los procesos mencionados en la sección anterior están ampliamente descritos en la presente memoria, mostrándose a continuación la relación de capítulos de la misma, y los aspectos que se tratan en cada uno de ellos:

- Parte I: *Preliminares*, en el que se realiza una introducción a los distintos campos y conceptos en base a los que se desarrolla el contenido de esta memoria.
- Parte II: *Análisis de microarrays mediante Clustering y Biclustering*, en la que se proponen y describen distintos algoritmos de clustering y biclustering para el

análisis de datos de microarrays. La primera aproximación se describe en el capítulo 2: *Clustering y biclustering para identificar patrones de máxima varianza en matrices de expresión genética*, y está basada en el Análisis de Componentes Principales y la optimización con Algoritos Evolutivos. La segunda aproximación, descrita en el capítulo 3: *Biclustering espectral posibilístico en matrices de expresión genética*, se basa en técnicas espectrales y Tecnología Difusa. En ambos capítulos se muestran los resultados de la aplicación de los algoritmos propuestos sobre microarrays reales de *S. Cerevisiae* y *H. Sapiens* y se realiza un análisis comparativo respecto a otros algoritmos clásicos de agrupamiento (clustering y biclustering), validando los resultados desde un punto de vista biológico utilizando la ontología *Gene Ontology* [32].

- Parte III: *Anotación manual y sistemas automáticos para la extracción de conocimiento de textos biomédicos.*, en la que se describen dos aproximaciones para la extracción de conocimiento de textos biomédicos de publicaciones científicas e historias clínicas, respectivamente. El capítulo 4: *Sistemas de anotación colaborativa de textos biomédicos*, describe el esquema para la anotación de un corpus especializado y propone una herramienta web de anotación colaborativa que permite integrar anotaciones humanas y automáticas para la creación de corpora de textos biomedicos. El Capítulo 5: *Identificación automática de diagnósticos en historias clínicas. Caso de estudio en obesidad.*, presenta un clasificador basado en regresión logística para la identificación automática del diagnóstico principal y comorbilidades en base al texto de historias clínicas. En este capítulo se muestran los resultados de la aplicación de la metodología propuesta sobre un corpus de historias clínicas para pacientes evaluados de obesidad y otras 15 comorbilidades relacionadas, se realiza un análisis detallado de las distintas estrategias propuestas y una comparativa de resultados respecto a otros algoritmos de clasificación.
- Parte IV: *Conclusiones*, en el que recogeremos las conclusiones principales obtenidas de la investigación llevada a cabo, y las líneas más prometedoras que se abren a partir de la misma.
- Parte V: *Publicaciones*, en el que se enumeran las publicaciones que recogen el trabajo presentado en esta memoria.
- Bibliografía: Referencias de las fuentes reseñadas en esta memoria.

Parte I

Preliminares

Capítulo

1

Preliminares

1.1 Introducción a la tecnología de microarrays de expresión.

En 1977 Fred Sanger y Alan R. Coulson publicaron una metodología para la secuenciación de cadenas de ADN que supuso una revolución en la investigación en biomedicina, al sentar las bases que permitirían secuenciar genes y, posteriormente, genomas completos [271]. Una década después, a finales de los años 80, los científicos Stephen Fodor, Michael Pirrung, Leighton Read y Lubert Stryer, desarrollaron una tecnología innovadora para la determinación y cuantificación del ADN de una muestra, tecnología que desembocaría posteriormente en la primera plataforma de microarrays de expresión.

Los microarrays de expresión surgen de la necesidad de analizar el enorme volumen de datos derivado de los proyectos de secuenciación de genomas [16, 17, 8]. Esta tecnología aporta, como principal ventaja frente a los métodos tradicionales, la alta densidad de integración de material biológico que se consigue inmovilizar, es decir, la posibilidad de analizar simultáneamente miles de genes en un único experimento [207]. Actualmente, esta tecnología se está aplicando, entre otros, a la identificación de perfiles genéticos y dianas terapéuticas, detección de mutaciones y polimorfismos, secuenciación, seguimiento de terapia, medicina preventiva, toxicología de fármacos y diagnóstico molecular.

Además de los microarrays de expresión, existen otras tecnologías de más reciente aparición y de prometedor futuro cuyo uso se está extendiendo actualmente en

la comunidad científica. Por ejemplo, las tecnologías de secuenciación de siguiente generación (*next-generation sequencing*) están revolucionando la biología actual, permitiendo reducir los costes y aumentar la velocidad de secuenciación de forma dramática respecto a los métodos basados en la metodología de Sanger [212]. La secuenciación de siguiente generación está siendo exitosamente aplicada a la secuenciación de genomas bacterianos para el estudio de la biodiversidad en distintos entornos, la secuenciación de genomas de especies extinguidas, y a la re-secuenciación (*resequencing*) de genomas utilizando secuencias de referencia de organismos enparentados, entre otros [278]. Otra tecnología en auge son los estudios de genoma completo (*genome-wide association studies*, GWA) [140] que utilizan tecnologías de genotipado de altas prestaciones para analizar genomas completos, identificar cientos de miles de polimorfismos de una sola base (*single-nucleotide polymorphisms*, SNPs) y ponerlos en relación con las condiciones clínicas de los individuos secuenciados. Existen numerosos trabajos de GWA que asocian SNPs con distintas enfermedades, priorizan los genes candidatos a causar una enfermedad, evidencian interacciones entre genes e identifican combinaciones de múltiples SNPs en un mismo gen que multiplican los riesgos de padecer determinadas patologías [244].

El desarrollo y aplicación de las mencionadas tecnologías de altas prestaciones en las ciencias de la vida, ha generado grandes masas de datos que es necesario procesar, analizar e interpretar. Ante esta necesidad, el estudio y desarrollo de nuevos métodos de Aprendizaje Automático y *Data-Mining*, enfocados al análisis de este tipo de datos, resulta fundamental para el avance de la genética y la biología molecular. En esta sección, se presentan los fundamentos biológicos en los que se basa la tecnología de microarrays en general, centrándose la atención posteriormente en los microarrays de expresión. También se describe el proceso experimental de preparación de muestras e hibridación de microarrays de expresión. Por último, se describe el proceso de análisis de los datos resultantes y se presentan las plataformas más populares de análisis de microarrays.

1.1.1 Microarrays. Fundamentos biológicos.

En los organismos eucariontes, el ADN está organizado en cromosomas. Los cromosomas son los portadores de la mayor parte del material genético, y condicionan la organización de la vida y las características hereditarias de cada especie. Éstos están compuestos de largas cadenas de ADN que contienen cientos de miles de genes que

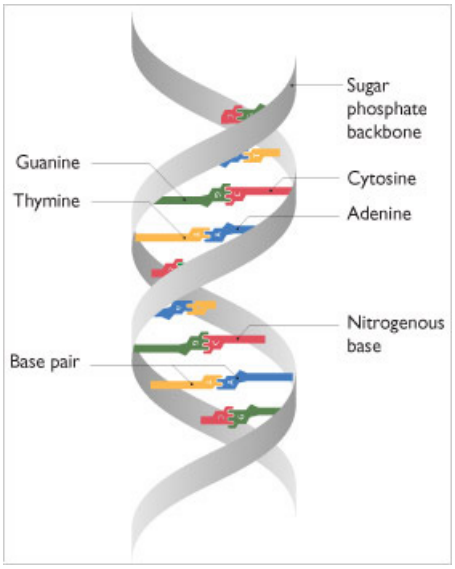


Figura 1.1: *Estructura de la doble hélice de ADN (http://www.wellcome.ac.uk/en/fourplus/-DNA.html)*

codifican la información necesaria para sintetizar proteínas.

Los ácidos nucleicos (ADN, *ácido desoxi-ribonucleico* y ARN, *ácido ribonucleico*) son polímeros de nucleótidos. Un nucleótido es una molécula formada por un azúcar (en el ADN la desoxi-ribosa y en el ARN la ribosa) y una base (en el ADN Guanina, Adenina, Citosina y Timina y en el ARN Guanina, Adenina, Citosina y Uracilo). Las bases se unen entre sí por medio de puentes de hidrógeno y permiten la asociación complementaria de dos cadenas de ácido nucleico.

Los patrones de formación de enlaces de hidrógeno fueron definidos por Watson y Crick en 1953, que establecieron que una Adenina se enlaza específicamente con una Timina (o un Uracilo en el ARN) y una Guanina se enlaza con una Citosina. Dos bases que pueden emparejarse se denominan complementarias (Adenina y Timina son complementarias, igual que Guanina y Citosina). Este emparejamiento de bases permite al ADN adoptar su estructura en doble hélice [327] y es la secuencia de bases emparejadas la que le permite codificar información y replicarla, utilizando cada cadena como molde para obtener una nueva cadena de bases, complementaria a la anterior (Figura 1.1). El descubrimiento de la estructura del ADN fue un acontecimiento trascendental para la ciencia y supuso el inicio de la era moderna de la bioquímica genética, era en la que el gen pasa a convertirse en la unidad fundamental de información en los sistemas vivos.

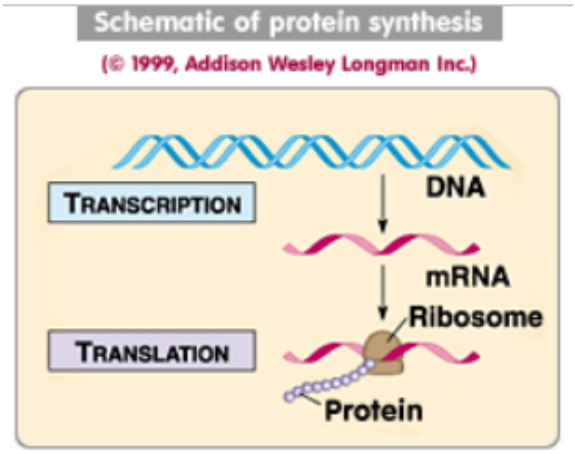


Figura 1.2: *Visión esquemática de la síntesis de proteínas.*

Un gen puede definirse, desde un punto de vista bioquímico, como un segmento único de ADN, de longitud variable, que codifica la información necesaria para la síntesis de una proteína.

El proceso de obtención de las proteínas a partir del genoma constituye lo que se denomina *dogma central de la biología molecular* y comprende los siguientes pasos (Figura 1.2):

1. Trascripción: obtención de ARN a partir de ADN.
2. Traducción: obtención de una proteína a partir de ARNm (ARN mensajero).

En la trascripción, la información representada por la secuencia de ADN es trasladada a una secuencia de ARN, proceso que se basa en la replicación del ADN. En la traducción, la información contenida en la secuencia de ARNm se emplea para la síntesis de proteínas, de acuerdo con una codificación en la que cada aminoácido que compone la proteína está representado por 3 bases de la secuencia de ARNm (la relación entre aminoácidos y los distintos tripletes de bases que los codifican se conoce como código genético).

La proteína es un componente estructural y funcional básico para la organización y funcionamiento celular. El proceso de síntesis de proteínas permite que la información genética almacenada en el ADN se convierta en proteínas. Todas las células de un organismo vivo poseen el mismo conjunto de cromosomas, y por lo tanto, pueden sintetizar el mismo repertorio de proteínas. La diferencia de concentración y abundancia de distintas proteínas permite que las células puedan exhibir distintas propiedades y comportamiento (especialización celular), a pesar de pertenecer al mismo individuo

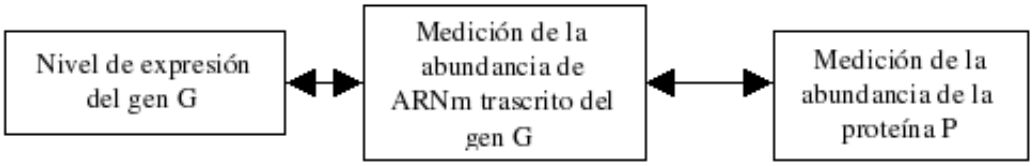


Figura 1.3: *Visión tradicional del dogma central de la biología molecular.*

y, por tanto, compartir el mismo genoma. Dado que la concentración de los distintos tipos de proteínas determina el comportamiento celular, resulta muy interesante desde el punto de vista biológico cuantificar las proteínas presentes en las células.

La tecnología de microarrays permite la detección y cuantificación de distintos tipos de moléculas para la caracterización celular. Según el tipo de molécula considerada (sonda), podemos distinguir distintos tipos de microarrays: de expresión (oligonucleótidos o ADNc), proteínas, carbohidratos, tejidos, etc. Nuestra investigación se centra en los microarrays de expresión.

Los microarrays de expresión son colecciones de moléculas de oligonucleótidos o ADNc (ADN complementario, molécula de ADN de una sola cadena) inmovilizadas en localizaciones conocidas sobre un soporte. En función del material biológico empleado como sonda, existen dos tipos fundamentales. Los microarrays de ADNc contienen secuencias completas de ADNc transcritas de los genes. Los microarrays de oligonucleótidos contienen secuencias de oligonucleótidos, habitualmente más cortas, que se corresponden con fragmentos de genes (*Expressed Sequence Tags*, EST) y son sintetizadas *in situ*. En el mercado existen diversas plataformas de microarrays de expresión, que son comercializadas por empresas como Affymetrix, Agilent, GE Healthcare, etc.

Los microarrays de expresión permiten cuantificar la presencia de miles de fragmentos de ARNm de secuencia conocida en las células de un individuo. La base teórica que sustenta la utilidad de los microarrays de expresión se corresponde con una visión tradicional del dogma central de la biología molecular, en la que se considera una relación uno-a-uno desde la secuencia de ADN (gen) a la secuencia de ARNm y a la proteína (Figura 1.3). Según esta visión, la medición de la abundancia de una determinada secuencia de ARNm nos ofrece una buena estimación del nivel de expresión del gen que se transcribe en dicha secuencia de ARNm (gen G en Figura 1.3) y de la abundancia de la proteína codificada con dicha secuencia (proteína P en Figura 1.3). Por tanto, el nivel de expresión de un gen en una determinada muestra puede

medirse calculando la abundancia de ARNm transcrito de dicho gen que está presente en las células de dicha muestra. Esto implica que existe una conexión entre el estado de actividad de una célula (el comportamiento de la célula) y la composición de su ARNm. Estas conexiones, que podrían ayudar a dar respuesta a un gran número de interrogantes en este campo (por ejemplo, explicar las causas genéticas de determinado comportamiento celular, o distinguir entre distintos estados celulares según la composición del ARNm), son las que se tratan de identificar en un microarray de expresión, en el que típicamente se monitorizan los niveles de expresión de secuencias de ARNm asociadas a miles de genes para decenas de muestras.

1.1.2 Tecnología de microarrays de expresión genética.

El objetivo de la tecnología de microarrays de expresión consiste en cuantificar la abundancia de miles de secuencias de ARNm (asociadas a los genes y fragmentos de genes -ESTs-) de una muestra biológica.

La Figura 1.4 muestra los pasos de un experimento de análisis de la expresión genética con microarrays de expresión. En este apartado se revisan los dos primeros pasos, relacionados con el diseño y elaboración del experimento de hibridación del microarray. La sección 1.1.3 describe las técnicas y plataformas más populares de normalización, análisis e interpretación de los datos de expresión.

Todo experimento de microarrays comienza con la definición del propósito científico (o hipótesis) del experimento. Además, resulta fundamental determinar el tipo de microarray (de cDNA u oligonucleótidos) y la plataforma comercial, así como las muestras o condiciones experimentales en estudio. Otro aspecto fundamental en el diseño del experimento de microarrays, es la elección de la tecnología de arrays de un color o dos colores [286]. La tecnología de un color (*single-color* o *single-channel*) mide los niveles de expresión de cada muestra en un microarray diferente. La tecnología de dos colores (*two-color* o *two-channel*) permite comparar niveles de expresión relativos entre una pareja de muestras en cada microarray. Los arrays de un color aportan más flexibilidad al análisis, mientras que los de dos colores permiten controlar algunas limitaciones técnicas permitiendo la comparación directa de muestras en un mismo experimento de hibridación [27]. Un estudio comparativo entre las tecnologías de un color y de dos colores, utilizando la misma plataforma, demuestra que existe una tasa de acuerdo alta entre los datos producidos por los dos tipos de tecnologías [243]. Los arrays de ADNc son típicamente de dos colores. Los arrays de oligonucleótidos

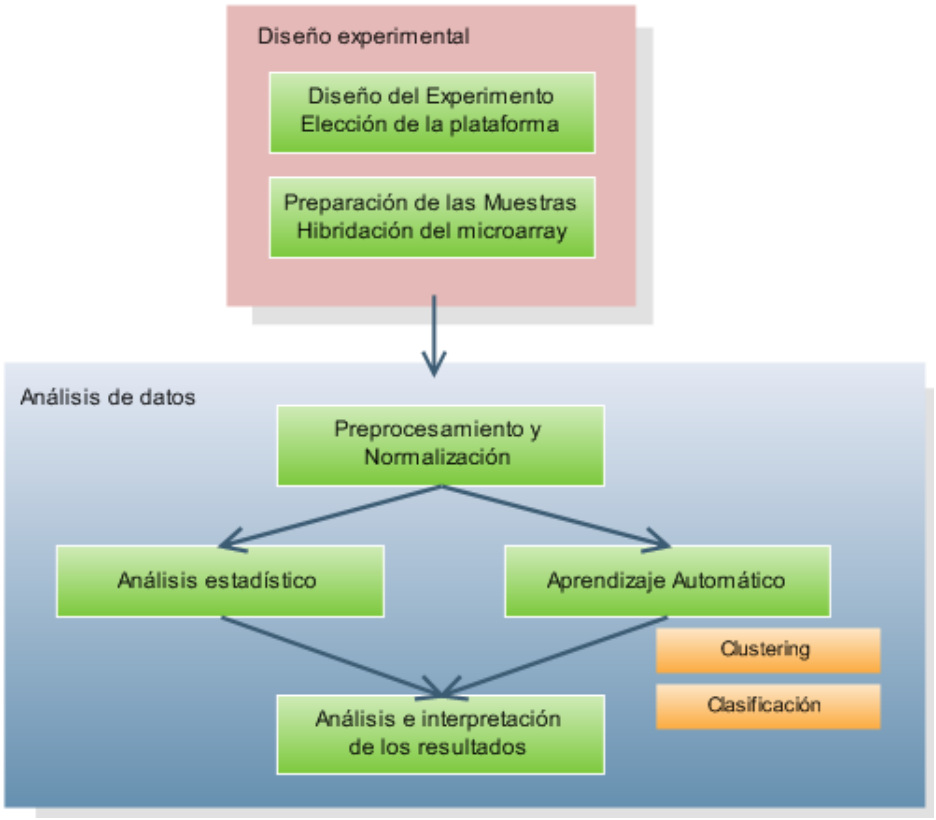


Figura 1.4: *Pasos de un experimento de análisis de microarrays de expresión.*

pueden ser de uno o dos colores (por ejemplo, Agilent y GE Healthcare comercializan arrays de uno o dos colores, mientras que Affymetrix emplea tecnología de un color).

El proceso de cuantificación de secuencias de ARNm en microarrays de expresión comprende los siguientes pasos:

1. Deposición e inmovilización de decenas de miles de sondas de ADNc u oligonucleótidos en posiciones (*spots*) conocidas y ordenadas de un soporte (habitualmente de plástico o vidrio).

2. Extracción de ARNm (*target* o diana) de una muestra. Debido a la inestabilidad del ARNm, es necesario realizar una transcripción inversa para obtener ADNc, que presenta mayor estabilidad. A continuación, se realiza un marcaje de la muestra con fluorocromos o biotina. Para microarrays de dos colores, se extrae ARNm de una muestra objetivo y una muestra referencia, realizándose el marcaje de cada una de ellas con fluorocromos distintos (típicamente rojo -Cy5- y verde -Cy3-, respectivamente).

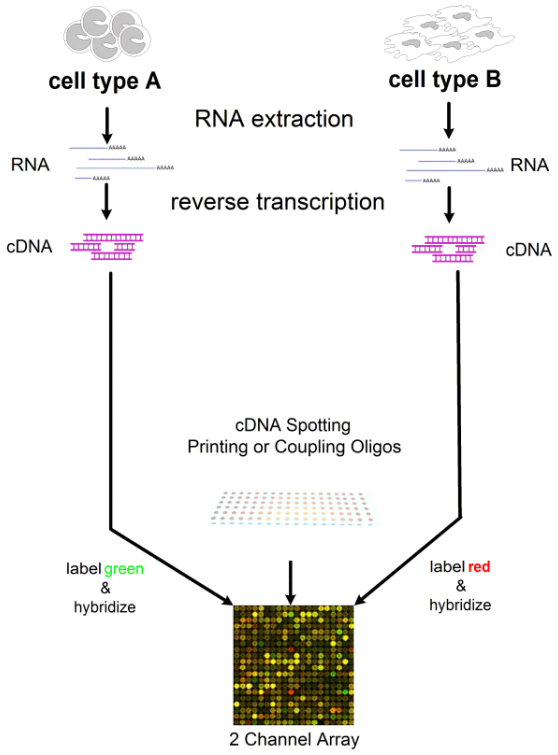


Figura 1.5: *Pasos para la hibridación en un microarray de expresión de ADNc.*

3. Las cadenas diana marcadas se ponen en contacto con el material genético depositado en el microarray, produciéndose la hibridación (por complementariedad de bases). En el caso de microarrays de dos colores, este proceso se denomina *hibridación competitiva*. Como resultado, cada *spot* del microarray adquiere un color e intensidad, en base al grado de hibridación de las secuencias del microarray con las cadenas diana marcadas.

4. Para medir el grado de hibridación, el microarray se excita mediante láser y se mide la intensidad y el color de la luz emitida por cada *spot*. En microarrays de un color, la intensidad de la luz indicará el nivel de expresión del gen en la muestra diana. En microarrays de dos colores, se mide además el color del *spot*, que tiende a rojo o verde dependiendo del tipo de muestra que más se haya hibridado con el contenido del *spot*, amarillo en caso de que las cantidades sean similares, o negro si no se ha producido hibridación con ninguna de las dianas.

El proceso completo de hibridación para arrays de dos colores se muestra en la Figura 1.5. La Figura 1.6 muestra un microarray de dos colores tras ser excitado con láser.

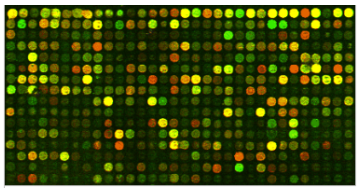


Figura 1.6: *Microarray de expresión de dos colores tras ser excitado con láser. En él se pueden apreciar los distintos colores e intensidades de luz de los spots, que revelan los niveles de expresión de los correspondientes genes en cada muestra.*

Cuando se desean estudiar muestras en distintas condiciones experimentales o de distintos pacientes, será necesario preparar e hibridar un microarray por cada una de las muestras de estudio. En el caso de arrays de dos colores, la muestra referencia suele consistir en un *pool* de material genético estándar.

1.1.3 Análisis de microarrays de expresión.

Una vez establecido el propósito científico (hipótesis), diseñado el experimento de microarrays (elección de plataforma y muestras) y llevado el mismo a cabo tal y como se describe en la sección anterior, las imagenes obtenidas son procesadas y la información resultante dispuesta en forma matricial, formando lo que se denomina matriz de expresión genética. La matriz de expresión genética (Figura 1.7) contiene el nivel de expresión de los genes (dispuestos en las filas de la matriz, un gen por cada fila) para cada una de las muestras de estudio (dispuestas en las columnas de la matriz, una muestra por columna).

El objetivo de nuestra investigación es el análisis de matrices de expresión para encontrar patrones de interés y asociaciones entre genes y muestras. Más detalladamente, la extracción de conocimiento de las matrices de expresión engloba los siguientes pasos (Figura 1.4):

1. Preprocesamiento y normalización de los datos.
   Antes de analizar los datos de las matrices de expresión es necesario normalizarlas, para eliminar la variabilidad debida a la tecnología e inherentemente ligada al proceso de hibridación de los arrays. La normalización de los datos de microarrays de expresión permite, fundamentalmente, corregir las diferencias en las distribuciones de los valores de intensidad de los distintos arrays. Según la plataforma y tecnología de microarrays empleadas, existen distintos métodos de normalización: *Global Mean/Median, Linear Lowess, Robust Linear*

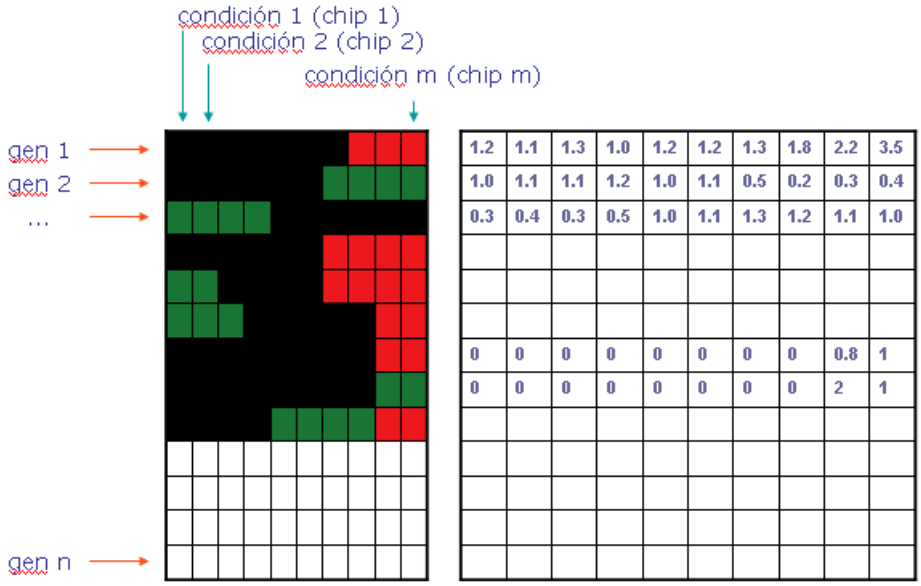


Figura 1.7: *Matriz de expresión genética obtenida de los experimentos de hibridación de microarrays.*

Lowess, Quadratic Loess, Robust Quadratic Loess, Rank Invariant, Quantile normalization, MAD centering, etc. Para una revisión completa, pueden consultarse [310, 55, 288, 257]. En esta fase también se incluyen tareas de procesamiento de los datos necesarias antes del análisis de los mismos, como son las siguientes:

- Transformación logarítima de escala. En arrays de ADNc el nivel de expresión de un gen se expresa como el cociente del valor de expresión en la muestra objetivo, respecto al valor de expresión en la muestra tomada como referencia. De este modo, un valor de expresión en el rango $[1, \infty)$ indica sobre-expresión del gen en la muestra objetivo respecto a la referencia, y un valor de expresión en el rango $[0, 1]$ indica sub-expresión del gen en la muestra objetivo respecto a la referencia. Para igualar las amplitudes de ambos rangos, se efectua una transformación logarítmica (habitualmente en base 2) de los datos de expresión relativa, que los traslada al rango $(-\infty, +\infty)$, con los valores positivos indicando sobre-expresión y valores negativos indicando sub-expresión.
- Tratamiento de réplicas. Habitualmente, un microarray inmoviliza varias sondas asociadas a un mismo gen. Para tratar los valores de expresión de estas sondas, primero se eliminan aquellos inconsistentes y posteriormente se agregan los valores del resto. Para la determinación de las réplicas

inconsistentes, se aplica un umbral de distancia máxima entre la réplica y la mediana de todas las réplicas. Todas aquellas réplicas que se hallen a una distancia mayor que el umbral elegido son eliminadas. El resto son promediadas utilizando la media o mediana.

- Tratamiento de valores perdidos. El proceso de hibridación del microarray y de medición de la intensidad de luz resultante puede fallar para algunos *spots* del microarray. En estos casos el valor desconocido se infiere utilizando algún método de recuperación de datos perdidos. Tras el trabajo pionero de Troyanskaya *et al.* [308], numerosos métodos han sido propuestos para la estimación de valores perdidos en microarrays [170, 312, 325]; aunque es el método de los K vecinos más cercanos (KNN o *K-nearest neighours*) el que, por su simplicidad, es empleado más habitualmente. Este método estima el valor perdido de un gen a partir de los valores, para esa misma muestra, de los genes con patrones de expresión más similares a dicho gen en la matriz de expresión. Para ello necesita: 1) un valor apropiado para K; 2) una medida de similitud para los genes (habitualmente se utiliza la distancia euclídea, correlación de Pearson, o minimización de la varianza); y 3) un método de agregación de los valores de los genes más cercanos para estimar el valor perdido (habitualmente la media o mediana de la componente desconocida para los K vecinos más cercanos).
- Eliminación de datos planos. Se eliminan los patrones que no presenten un número determinado de valores que superen, en valor absoluto, un valor umbral.

Existen numerosas *suites* de software de normalización y análisis de microarrays que implementan distintos métodos de preprocesamiento y normalización para distintas plataformas[301, 102]. La tabla 1.1 lista algunas de estas herramientas.

2. Análisis y modelado de los datos.
   En esta categoría pueden encuadrarse las técnicas estadísticas y de Aprendizaje Automático que se aplican sobre los datos ya preprocesados para la extracción de conocimiento (Figura 1.4). Las técnicas estadísticas son habitualmente empleadas como primera aproximación al análisis de microarrays de expresión, habiéndose desarrollado técnicas específicas para el análisis de este tipo de datos, como SAM (*Significance Analysis of Microarrays*) [315]. Algunas de estas técnicas han sido aplicadas al análisis de datos de microarrays de expresión

proporcionados por el Hospital Virgen de las Nieves de Granada, dando lugar a algunos trabajos publicados y referenciados en la Sección 8. Mientras que las técnicas estadísticas proporcionan una buena primera aproximación al análisis de arrays de expresión, resultan limitadas por el volumen y naturaleza de los datos. De este modo, se hace necesaria la aplicación de técnicas de Aprendizaje Automático para analizar este tipo de datos. En particular, resultarán de especial interés los algoritmos de Agrupamiento (Clustering y Biclustering), para los que se propone una revisión en la Sección 1.2.

3. Interpretación y validación de los resultados.
   Algunas de las técnicas más extendidas para la interpretación y validación de los resultados son la validación cruzada, los tests estadísticos, la inspección visual de los resultados y la validación biológica de los mismos respecto al conocimiento existente y a nuevos experimentos. En última instancia, será la facilidad de interpretar biológicamente los resultados y su adecuación a la realidad, lo que determinará la bondad y corrección de los experimentos y técnicas empleadas en fases anteriores.

La Tabla 1.1 muestra un listado de las plataformas más populares de normalización, análisis e interpretación de las matrices de expresión genética obtenidas de microarrays.

1.1.4 Aplicaciones de los microarrays.

Con un creciente número de genomas secuenciados, son numerosas las cuestiones relacionadas con la comprensión de los procesos biológicos celulares y la regulación genética que están recibiendo la atención de la comunidad científica. Algunas de estas cuestiones son las siguientes:

- ¿Cuál es la función de los distintos genes y en qué procesos participan?
- ¿Cómo se regulan los genes? ¿Cómo interaccionan los genes?
- ¿Qué genes son responsables de una determinada enfermedad o comportamiento anómalo celular?
- ¿Cómo cambia el nivel de expresión de ciertos genes cuando las células están sometidas a un determinado tratamiento?

La demanda de aplicaciones y métodos que arrojen luz sobre estas preguntas ha sido espectacular en los últimos años. Las aplicaciones de los microarrays de expresión se reparten en campos tan amplios como el desarrollo agrícola, la ingeniería genética

Normalización y análisis de microarrays de expresión	
Bioconductor	http://www.bioconductor.org
GEPAS	http://gepas.bioinfo.cipf.es/
WebArrayDB	http://www.webarraydb.org
MayDay	http://www-ps.informatik.uni-tuebingen.de/mayday/wp/
MDAT (Matlab toolbox)	http://www.mathworks.com/matlabcentral/fileexchange/5037-mdat
Gene ARMADA	http://www.grissom.gr/armada/
ArrayNorm	http://genome.tugraz.at/arraynorm/arraynorm_description.shtml
TM4 / MeV	http://www.tm4.org/mev/
PreP+07	http://www.bitlab-es.com/prep
FlexArray	http://genomequebec.mcgill.ca/FlexArray/
EMMA	http://www.cebitec.uni-bielefeld.de/groups/brf/software/emma_info/index.html
Análisis y anotación funcional	
DAVID	http://david.abcc.ncifcrf.gov/
Babelomics	http://www.babelomics.org/
Gene Set Enrichment Analysis (GSEA)	http://www.broadinstitute.org/gsea/

Cuadro 1.1: *Principales plataformas software para análisis de datos de microarrays.*

o la salud humana, entre otros. En el sector de la salud humana, las aplicaciones van, siguiendo las directrices marcadas por las preguntas anteriores, desde la detección de variaciones y mutaciones en el genoma, hasta la identificación de la funcionalidad de cada gen, pasando por la caracterización del perfil genético de enfermedades, diagnóstico molecular, farmacogenómica y un largo etcétera.

Podemos clasificar estas aplicaciones en las siguientes categorías [207]:

A) Monitorización de la expresión genética.

El patrón de expresión de un gen proporciona información acerca de su comportamiento en distintas muestras o condiciones experimentales. Así, la monitorización y análisis de los patrones de expresión genética han sido utilizados, desde la aparición de la tecnología de microarrays, con el fin de identificar cuáles son las funciones de los distintos genes y en qué procesos celulares participan.

B) Generación de nuevos fármacos.

Las aplicaciones de microarrays contribuyen a la identificación de dianas terapéuticas más efectivas, como parte de las primeras etapas del proceso de in-

vestigación en la industria farmacéutica y de generación de nuevos fármacos. La identificación del gen responsable de un determinado mecanismo biológico implicará la aparición de una nueva diana terapéutica, de forma que se desarrollarán nuevos fármacos que actúen sobre la misma para la regulación de dicho mecanismo biológico. Los arrays de expresión son una herramienta potencial para la investigación de los mecanismos por los cuales un fármaco actúa, asi como su implicación en rutas metabólicas y posibles efectos secundarios.

C) Farmacogenómica.

La farmacogenómica se define como el estudio del impacto que presentan las variaciones genéticas en la eficacia y toxicidad de los fármacos. También se define como el estudio de cómo el componente genético de un paciente determina la respuesta frente a una terapia determinada. Los estudios farmacogenómicos correlacionan el perfil genético de los individuos con la respuesta de cada uno de ellos a un fármaco determinado. La información obtenida con estos resultados se utiliza para diseñar arrays de expresión que puedan ser utilizados en la selección de fármacos a medida (medicina personalizada), el diseño de tratamientos más eficaces y la atenuación de posibles efectos secundarios en el paciente.

D) Diagnóstico molecular.

Mediante las técnicas apropiadas, el análisis de los microarrays de expresión puede conducir a la identificación de marcadores específicos para determinadas enfermedades, es decir, a la caracterización de enfermedades a través del reconocimiento de ciertos patrones de expresión para determinados genes. Esto permitiría un diagnóstico más rápido de la enfermedad y abriría posibles nuevas vías de tratamiento.

1.1.5 Limitaciones de los microarrays de expresión genética.

1.1.5.1 Limitaciones conceptuales.

En la sección 1.1.1, para justificar biológicamente porqué a partir de microarrays de expresión puede extraerse conocimiento útil acerca del estado funcional celular, nos amparábamos en la visión tradicional del dogma central de la biología molecular, que consideraba una relación uno-a-uno desde la secuencia de ADN (gen), a la secuencia de ARNm, a la proteína. De esta forma, la medición de la abundancia del ARNm transcrito de un determinado gen, nos proveía de una aproximación del nivel de

expresión de dicho gen en la muestra, de la abundancia de la proteína codificada en dicho gen y, por ende, del comportamiento de la célula de la muestra estudiada.

Desafortunadamente, el proceso de expresión genética es más complejo. La visión moderna del dogma central de la biología molecular contempla un escenario más dinámico, según el cual un gen puede codificar más de una secuencia de ARNm. En el proceso de síntesis de ARNm a partir de la secuencia de ADN asociada a un gen, el proceso de trascripción usa un mecanismo denominado *splicing* (corte y empalme) para descartar ciertas regiones de la cadena de ADN (denominadas intrones), de forma que sólo los segmentos no eliminados (denominados exones) participan en la generación de la secuencia de ARNm y, por tanto, en la codificación de la proteína (Figura 1.8). Además, en el proceso de *splicing* no siempre se seleccionan los mismos exones (*alternative splicing*). Podemos considerar los intrones como regiones del gen que, en teoría, no codifican información, ya que aunque la información genética se encuentra almacenada en el ADN, no todas las secuencias de ADN son codificantes. Una secuencia codificante (exon) está formada por una sucesión de nucleótidos que, leídos de 3 en 3 y de manera no solapada, codifican una secuencia de aminoácidos, incluyendo, además, las señales de iniciación y terminación en ambos extremos. En el ADN humano, la mayor parte del ADN total es no codificante. Actualmente existen un amplio campo de estudio que tiene por objeto la identificación de secuencias de ADN (denominadas *genes RNA*) que codifican moléculas funcionales de RNA que no son traducidas en proteínas, denominadas RNAs no codificantes (*non-coding RNAs*, ncRNA). Estas moléculas de ncRNAs desempeñan un papel importante en la regulación genética, ya que habitualmente se corresponden con moléculas de RNA de transferencia (tRNA), RNA ribosómico (rRNA), u otros tipos de RNA como snoRNAs, microRNAs, siRNAs, piRNAs, entre otros [1].

De este modo, distintos particionamientos o splicings sobre un mismo gen pueden producir distintas moléculas de ARNm, que pueden codificar distintas proteínas. Por ejemplo, se ha observado que un gen expresado en el cerebro de la mosca de la fruta (*Drosophila melanogaster*), puede ser particionado de 40.000 formas diferentes [72].

Otras fuentes de variación se deben a los factores de transcripción que desencadenan la transcripción de un gen, o los sitios de unión (*binding sites*) de la región *upstream* (cadena arriba, antes de la región codificante) del gen a los que se une un determinado factor de transcripción. Distintos factores de transcripción pueden producir la transcripción de un determinado gen (en ocasiones, es la combinación de

[1] `http://en.wikipedia.org/wiki/Non-coding_RNA` y `http://en.wikipedia.org/wiki/List_of_RNAs`

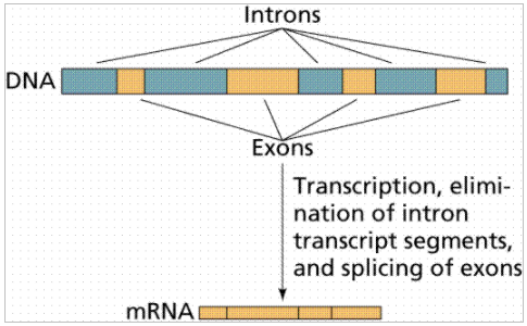


Figura 1.8: *El papel de los intrones y exones en la transcripción. Imagen de An On-Line Biology Book (http://www.emc.maricopa.edu/faculty/farabee/BIOBK/BioBookTOC.html).*

distintos factores la que desencadena el proceso). Además, estos factores de transcripción pueden unirse a distintos sitios de unión de los genes. Cada factor de transcripción y/o sitio de unión al que se acople pueden producir la transcripción de una cadena diferente de ARNm.

También es posible -se conocen algunos casos y desconocemos cuántos más pueden darse- que se produzca la "edición" de una secuencia de ARNm antes de la traducción: una base de la secuencia puede ser cambiada por otra, modificándose así el aminoácido codificado con el triplete de bases afectado, o incluso convirtiendo un triplete que codifica un aminoácido en un triplete de terminación, finalizando antes la traducción, generándose en cualquiera de los dos casos una proteína distinta.

Además, la tasa de traducción es variable, es decir, puede que no se produzca la traducción siempre con la misma frecuencia ni velocidad, así que aunque en general existe una correlación entre la abundancia de ARNm y la abundancia de la proteína correspondiente, estas tasas de abundancia pueden presentar descompensaciones o parecer incorreladas en determinados momentos.

También pueden producirse modificaciones en la estructura de la proteína después de la traducción. La proteína puede ser "troceada" en otras de menor tamaño, o algunas de sus regiones de aminoácidos pueden ser eliminadas o modificadas por adición (temporal o permanente) de otros grupos químicos, a veces con efectos drásticos en las propiedades funcionales o estructurales de la proteína, y por tanto en el comportamiento celular.

Con todos estos inconvenientes, hemos de tener presente que la tecnología de microarrays de expresión mide la abundancia de ARNm trascrito, que como se ha comentado no tiene porqué corresponder exactamente con el nivel de expresión del gen asociado, ni con la abundancia de la proteína que codifica dicho gen. Dado que

los niveles de ARNm pueden no reflejar los niveles de proteína, y que la expresión de una proteína no siempre tiene una consecuencia fisiológica, sería necesario utilizar una técnica de análisis con indicadores más sofisticados, como la localización de las proteínas y sus tasas de recambio, cambios estructurales y modificaciones de proteínas. Los chips de proteínas de alta densidad de integración han surgido recientemente como una tecnología complementaria a los microarrays de expresión y presentan un enorme potencial. Estos microarrays sufrirán una rápida expansión debido al crecimiento del campo de la Proteómica en los últimos años [208].

1.1.5.2 Limitaciones técnicas.

Además de las limitaciones mencionadas anteriormente, existen numerosas fuentes de error debidas a imprecisiones e imperfecciones en el instrumental técnico utilizado para la hibridación y escaneado de los microarrays de expresión. Algunos de estos errores son, por ejemplo, desviaciones en la cantidad de material biológico impreso en cada *spot* del microarray o en la cantidad de reactivo fluorescente con el que se marcan las muestras, errores en la medición de luz por parte del escáner, etc.. Estos errores son inevitables, y es necesario manejarlos para que el posterior análisis y las conclusiones que se extraigan del mismo sean fiables. El preprocesamiento y la normalización de los microarrays de expresión previo al análisis es, por lo tanto, fundamental.

A pesar de las limitaciones comentadas, la tecnología de microarrays de expresión resulta relativamente barata y permite una alta densidad de integración, con la medición en paralelo de los niveles de expresión de miles de genes en decenas de muestras. Constituye, por tanto, una herramienta útil con la que extraer información acerca de los procesos celulares, a pesar de que los resultados obtenidos del análisis de microarrays deban ser interpretados con cuidado y validados con información procedente de otras fuentes de datos.

1.2 Aprendizaje automático y bioinformática

La Bioinformática puede considerarse como un pilar imprescindible en los proyectos de genómica y proteómica en los que es necesario organizar resultados, analizarlos, generar hipótesis y proponer nuevos experimentos. Esta actividad ha hecho que la Bioinformática se convierta en un componente básico para el desarrollo de la biología molecular, la biotecnología y la biomedicina.

La Bioinformática puede definirse como la disciplina que tiene por objeto la investigación, desarrollo y aplicación de enfoques y herramientas computacionales para adquirir, almacenar, organizar, analizar e interpretar datos de tipo biomédico. Estos datos, derivados principalmente de la secuenciación de genomas y de la investigación experimental, están distribuidos en centenares de repositorios y bases de datos, que contienen información altamente especializada de composición muy heterogénea (ver Tabla 1.2).

El enorme incremento en la cantidad y complejidad de datos biológicos disponibles, ha motivado la necesidad de herramientas y técnicas capaces de analizar estos datos y extraer conocimiento novedoso y útil de los mismos. Para satisfacer esta necesidad, el Aprendizaje Automático (*Machine Learning*, ML) se ha erigido como una disciplina fundamental en centros de investigación biomédicos [190].

Según [224], se dice que un programa de ordenador aprende de una experiencia E disponible respecto a la realización de una tarea T y una medida de evaluación P, si su rendimiento, medido por P, para la realización de la tarea T, mejora utilizando la experiencia E.

El Aprendizaje Automático puede definirse como la disciplina de la Inteligencia Artificial (*Artificial Intelligence*, AI), dedicada al diseño de algoritmos para identificar regularidades, patrones o reglas en un conjunto de datos disponibles y representar el *conocimiento* adquirido para modelar y explicar los datos y para efectuar predicciones, diagnósticos, reconocimiento, controles, validaciones o simulaciones [185]. Las técnicas de aprendizaje automático se han aplicado exitosamente en una amplia variedad de aplicaciones bioinformáticas [190, 152], y son ampliamente utilizadas para investigar los mecanimos que subyacen en los procesos de regulación genética y para el descubrimiento de biomarcadores.

Un campo íntimamente relacionado con el aprendizaje automático, que también resulta de interés para el contenido de esta memoria, es la Minería de Datos (*Data Mining*, DM). En [108] se define la Minería de Datos como un proceso no trivial de identificación de patrones válidos, novedosos, potencialmente útiles y compresibles

Secuencias de ADN	
EMBL-Bank	http://www.ebi.ac.uk/embl/
GenBank	http://www.ncbi.nlm.nih.gov/entrez/query.fcgi?db=Nucleotide
RefSeq	http://www.ncbi.nlm.nih.gov/RefSeq/
DDBJ	http://www.ddbj.nig.ac.jp/
Ensembl	http://www.ensembl.org/
Secuencias de proteínas	
SwissProt	http://us.expasy.org/sprot/
Uniprot	http://www.ebi.ac.uk/uniprot/
Estructuras de proteínas	
PDB	http://www.rcsb.org/pdb/
Variabilidades génicas	
dbSNP	http://www.ncbi.nlm.nih.gov/projects/SNP/
Literatura científica	
PubMed	http://www.ncbi.nlm.nih.gov/entrez/query.fcgi?db=PubMed
PubMedCentral	http://www.ncbi.nlm.nih.gov/pmc
MeSH	http://www.ncbi.nlm.nih.gov/mesh
Arrays de ADN	
SMD	http://genome-www5.stanford.edu/
ArrayExpress	http://www.ebi.ac.uk/arrayexpress/
Gene Expression Omnibus (GEO)	http://www.ncbi.nlm.nih.gov/gds
Interacción de proteínas	
PSI standard	http://psimi.ibioinformatics.org
Intact	http://www.ebi.ac.uk/intact/index.html
Mint	http://mint.bio.uniroma2.it/mint/
Bases de datos Médicas	
OMIM	http://www.ncbi.nlm.nih.gov/entrez/query.fcgi?db=OMIM
Bases de datos de genes	
GeneCards	http://bioinfo.weizmann.ac.il/cards/
UniGene	http://www.ncbi.nlm.nih.gov/UniGene/
Ontologías	
Gene Ontology	http://www.geneontology.org
Human Disease Ontology (DOID)	http://diseaseontology.sourceforge.net/
Gene Regulation Ontology (GRO)	http://www.ebi.ac.uk/Rebholz-srv/GRO/GRO.html
BioPortal	http://bioportal.bioontology.org/ontologies
Rutas metabólicas	
KEGG	http://www.genome.ad.jp/kegg/
BioCYC	http://biocyc.org

Cuadro 1.2: *Principales bases de datos utilizadas en biología molecular.*

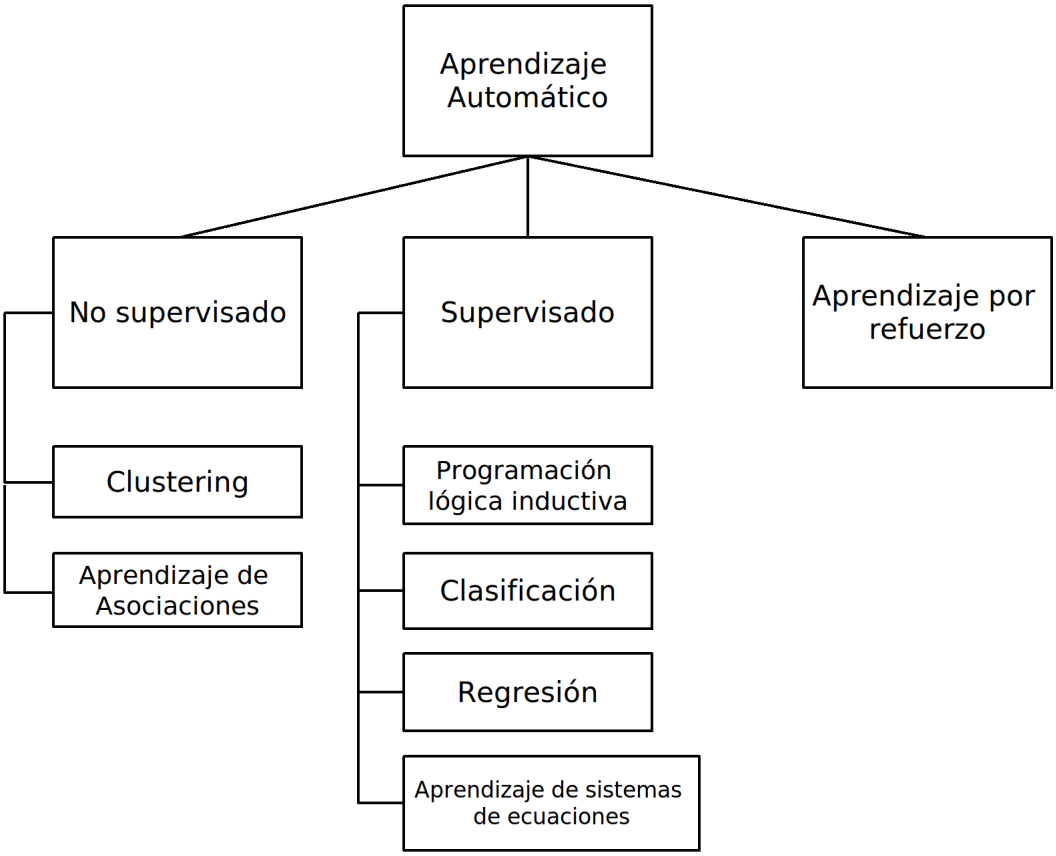


Figura 1.9: *Taxonomía de métodos de aprendizaje automático [185].*

en los datos. En [138] se añade que la tarea fundamental de la Minería de Datos es "encontrar modelos inteligibles a partir de los datos", y sintetiza sus objetivos en, por un lado, "trabajar con grandes volúmenes de datos" y por el otro "usar técnicas adecuadas para analizar los mismos y extraer conocimiento novedoso y útil". En el presente trabajo de investigación proponemos diversos métodos de aprendizaje automático para resolver tareas de data mining sobre grandes masas de datos biológicos.

La Figura 1.9 muestra una taxonomía de métodos de aprendizaje automático [185], en la que pueden distinguirse tres grandes grupos:

- Aprendizaje supervisado (*supervised learning*). Las técnicas de aprendizaje supervisado requieren que se posea cierto conocimiento *a priori* sobre los datos. Supongamos que un conjunto de datos D se compone de *objetos* $D = \{x_i \mid i = 1..N\}$, cada uno descrito mediante un conjunto de atributos o *características* $x_i = \{x_{ij} \mid j = 1..p\}$. Las características x_{ij} son variables independientes, contínuas o discretas. Del mismo modo, cada objeto x_i puede ser asociado a una

variable y, denominada comúnmente *clase* o variable objetivo. El aprendizaje supervisado tiene como objetivo la identificación de una función de mapeo $f : f(x_i) \rightarrow y$. Para identificar la función f, las técnicas de aprendizaje supervisado hacen uso de un conjunto de datos, que notamos como D_T, para los que se conoce su valor de y (objetos *etiquetados*): $D_T = \{(x_i, y) \mid i = 1..N\}$. D_T conforma lo que se denomina el *conjunto de entrenamiento*. Dentro del aprendizaje supervisado podemos encuadrar las técnicas de clasificación, regresión, el aprendizaje de sistemas de ecuaciones, y la programación lógica inductiva (*inductive logic programming*, ILP) [185].

- Aprendizaje no supervisado (*unsupervised learning*). El aprendizaje no supervisado no requiere de información *a priori* sobre los datos, es decir, se desconoce el valor de la variable objetivo o clase y de cada objeto. El clustering es la técnica más importante de aprendizaje no supervisado.
- Aprendizaje por refuerzo (*reinforcement learning*). El aprendizaje por refuerzo aborda el problema de enseñar a un agente autónomo a seleccionar las acciones óptimas para alcanzar un objetivo. El agente recibe información del estado actual del entorno en el que se encuentra y desempeña acciones para cambiarlo. Cada vez que el agente realiza una acción, recibe un refuerzo positivo (recompensa) o negativo (castigo).

Otro tipo de técnicas de ML que no se encuadran en las taxonomías tradicionales por su más reciente aparición y extensión, son las técnicas "Semi-supervisadas" de aprendizaje. Estas técnicas hacen uso de datos etiquetados y no etiquetados para el entrenamiento [348] y suelen emplearse en campos en los que se dispone de una pequeña cantidad de datos etiquetados (principalmente debido al alto coste o dificultad para obtenerlos) y, al mismo tiempo, de una gran cantidad de datos no etiquetados. Un ejemplo de este tipo de dominios puede ser el análisis de datos de expresión genética para encontrar módulos funcionales (se conoce la función de unos pocos genes) o la identificación de entidades o relaciones de interés en textos biomédicos (los corpora de anotaciones disponibles son de pequeño tamaño).

Según el tipo de criterio de optimización empleado por el algoritmo de ML, podemos clasificar los problemas de ML en *problemas de modelado* y *problemas de optimización*. En los problemas de modelado, el criterio de optimización es la bondad de ajuste de un modelo predictivo. En los problemas de optimización, el criterio a optimizar es una función de evaluación o *fitness* [190]. Los problemas de modelado tienen como objetivo la inferencia de un modelo que explique los datos o permita efectuar predic-

ciones. Para la inferencia de estos modelos, los métodos de ML se basan en técnicas estadísticas y probabilísticas. Algunas de las técnicas de ML asociadas a la inferencia de modelos son los modelos ocultos de Markov (*Hidden Markov Models*, HMM), las redes bayesianas, la regresión logística (*logistic regression*, LR), los árboles de decisión o las máquinas de vector soporte (*support vector machines*, SVM), entre otros.

Los problemas de optimización tienen como objetivo la identificación de una solución óptima en un espacio de múltiples soluciones posibles (habitualmente de enorme tamaño). Las técnicas de optimización pueden, a su vez, ser clasificadas en métodos exactos y aproximados, en función del tipo de solución que identifican. Los métodos exactos proporcionan la solución óptima, siempre que exista convergencia. Los métodos aproximados siempre proporcionan una solución candidata al óptimo, pero sin garantizar que sea la óptima. Los métodos de Monte-Carlo, el enfriamiento simulado, la búsqueda tabú y los algoritmos evolutivos son algunos de los métodos aproximados de optimización que han sido ampliamente utilizados en bioinformática [190]. De particular interés para el trabajo de investigación presentado en esta memoria resultan los algoritmos genéticos y los algoritmos de estimación de distribuciones de probabilidad (*Estimation of Distribution Algorithms*, EDAs), dos tipos de algoritmos evolutivos que se describen en detalle en el capítulo 2.

A continuación, se presenta una introducción más detallada a las técnicas de aprendizaje automático utilizadas en el presente trabajo de investigación. La primera parte de la memoria se centra en el estudio de técnicas de aprendizaje no supervisadas, en particular de clustering y biclustering, para la identificación de patrones de interés en el análisis de matrices de expresión genética. La segunda parte de la memoria presenta la utilización de técnicas de aprendizaje supervisado, en particular de clasificadores de regresión logística, para la identificación automática de diagnósticos en historias clínicas.

1.2.1 Clustering

El clustering es el método más extendido de aprendizaje no supervisado. Podemos definir el clustering como una técnica para dividir un conjunto de datos en grupos de objetos similares. En cada grupo, también denominado *cluster*, se incluirán objetos similares entre sí y distintos de los objetos de otros grupos. El objetivo es maximizar la similitud entre los elementos de un grupo y minimizar la similitud entre los distintos grupos. De esta forma, el objetivo del clustering es encontrar estructuras intrínsecas a un conjunto de objetos. Históricamente, el problema del agrupamiento de datos ha

sido abordado en numerosos contextos y áreas de aplicación, y su utilidad como un paso clave en el análisis exploratorio de datos es manifiesta [185].

Para determinar la (dis)similitud entre objetos y, de este modo, la bondad de las estructuras encontradas, es necesario definir una función de similitud o distancia que se adecúe al contexto y que produzca grupos de interés. Algunas de las medidas de similitud o distancia entre objetos, más comúnmente utilizadas por los métodos de clustering, son la distancia Minkowski, Euclídea, Manhattan, Mahalanobis, la correlación de Pearson o la separación angular (todas ellas descritas ampliamente en la literatura [185, 332]).

En ocasiones, los métodos de clustering también necesitan medir la similitud o distancia entre grupos de objetos (clusters). Las dos aproximaciones más extendidas para medir la distancia entre grupos consisten en: 1) agregar las distancias entre los elementos de los dos grupos; 2) calcular las distancias entre grupos en base a la elección de un objeto representante del grupo. Por ejemplo, si se toma la distancia mínima entre dos objetos, uno de cada grupo, decimos que se considera la distancia al vecino más cercano (*nearest neighbour dissimilarity*), que es la base de los algoritmos denominados *single linkage clustering*. Si se toma la distancia máxima entre dos objetos, uno de cada grupo, decimos que se considera la distancia al vecino más lejano (*furtherst neighbout dissimilarity*), que es la base de los algoritmos denominados *complete linkage clustering*. Si, en lugar de tomar los casos extremos, se utiliza una medida de distancia entre grupos en base a la distancia promedio entre objetos de ambos grupos, contamos con la base para aplicar un algoritmo de clustering denominado *average linkage clustering*.

Los algoritmos de clustering pueden clasificarse atendiendo a distintos criterios:

A) Exclusivos o no exclusivos. Los algoritmos de clustering exclusivos agrupan los objetos de forma exclusiva, es decir, si un objeto se asigna a un grupo, no puede ser asignado a ningún otro grupo. Por otro lado, los algoritmos de clustering no exclusivos permiten que un mismo objeto sea asignado a más de un grupo, por lo que los grupos pueden solaparse.

B) Jerárquicos o particionales. Los algoritmos de clustering jerárquicos unen, paso a paso, grupos ya formados, o bien dividen grupos en otros de menor tamaño. De este modo, un algoritmo de clustering jerárquico genera una secuencia de grupos anidados. Según la filosofía que se emplee en la obtención de esta secuencia de clusters, distinguiremos dentro de los algoritmos jerárquicos en-

tre algoritmos aglomerativos (incrementales o *bottom-up*) y algoritmos divisivos (decrementales o *top-down*):

- Los algoritmos jerárquicos aglomerativos parten de grupos conformados por un único objeto y combinan, sucesivamente, los grupos más similares en grupos de mayor tamaño.
- Los algoritmos jerárquicos divisivos parten de un único grupo que contiene todos los objetos y lo dividen, sucesivamente, en grupos de menor tamaño.

En ambos casos, el resultado final del clustering jerárquico es un árbol de clusters, cuya representación gráfica, denominada dendrograma, muestra cómo están relacionados los clusters que se han obtenido en los sucesivos pasos. *Cortando* el dendrograma en un determinado nivel, se obtiene una partición del conjunto inicial de objetos en un conjunto de grupos disjuntos. Algunos de los algoritmos de clustering jerárquico más populares son el *single linkage clustering*, *complete linkage clustering*, *average linkage clustering*, *balanced iterative reducing and clustering using hierarchies* (BIRCH), *clustering using representatives* (CURE), *robust clustering using links* (ROCK), todos ellos de tipo aglomerativo; y *divisive analysis* (DIANA) o *monothetic analysis* (MONA), de tipo divisivo [332].

Los algoritmos de clustering particional generan una partición del conjunto de objetos en un número determinado de grupos, optimizando una función criterio (en general, maximizando la similitud entre patrones asociados a un mismo grupo y minimizando la similitud entre grupos). En este caso, por lo tanto, no se obtienen secuencias de grupos anidados ni dendrogramas, característicos del clustering jerárquico. Algunos de los algoritmos particionales de clustering más extendidos son el K-medias (*K-means*), y el PAM (*Partitioning Around Medioids*) [185].

En particular, el algoritmo K-medias resulta de especial interés por su utilización en capítulos posteriores de esta memoria y se presenta a continuación.

1.2.1.1 K-medias.

Sea $X = \{x_i \mid i = 1..N\}$, con $x_i \in \Re^p$, el conjunto de N objetos p-dimensionales que se desean agrupar. Suponemos que X tiene una estructura de k clusters, es decir, cada objeto de X es asignado a una y sólo una de las k particiones. Cada cluster se representa por un único punto de $\Re^p$ que denominamos *prototipo* o *centroide*. Notamos como $L = \{l_i \mid i = 1..k\}$ la k-tupla de centroides, donde l_i representa el centroide del

cluster i. $M = \{m_i \mid i = 1,..,N\}$ es el vector de longitud N que indica qué cluster se asigna a cada objeto, donde m_i indica la etiqueta del cluster $\{1..k\}$ asignado al objeto x_i. Finalmente, notamos como d la función de distancia entre objetos.

Algoritmo K-medias

INPUT: Conjunto de datos X y número k de clusters de X.

OUTPUT: Conjunto de centroides L de los k clusters y vector M que indica el cluster asignado a cada objeto de X.

1. Inicializar el conjunto de centroides L escogiendo k objetos al azar de X.
2. Repetir:
   a) Reasignar los objetos de X al cluster j de centroide l_j más cercano: $argmin_j\, d(x_i, l_j)^2$.
   b) Actualizar M de forma que m_i es la etiqueta del cluster asignado a x_i.
   c) Recalcular los centroides L tal que l_j es el promedio de los objetos asignados al cluster j.
3. Hasta que converja, según la función objetivo $\sum_{i=1}^{N}(argmin_j\, d(x_i, l_j)^2)$

El clustering particional resulta más apropiado para representaciones eficientes y análisis de grandes masas de datos: los dendrogramas del clustering jerárquico son prácticamente inviables y pierden todo su poder expresivo con unos pocos cientos de objetos. El principal inconveniente del clustering particional es que suele ser necesario dar un valor *a priori* del número de clusters que se desean encontrar. La estimación de este valor es compleja y existen numerosos métodos computacionales diseñados a tal efecto [306, 222]. En este sentido, los dendrogramas proporcionados por el clustering jerárquico, siempre que el conjunto de objetos no sea demasiado numeroso, pueden facilitar la inspección visual de los agrupamientos formados, y la elección de un nivel en el que los agrupamientos parezcan recoger, con mayor bondad, la estructura de los objetos.

1.2.2 Clustering difuso.

En 1965 Lofti A. Zadeh introdujo la Teoría de los Conjuntos Difusos para representar formalmente la imprecisión intrínseca a ciertas clases de objetos [340]. Por ejemplo, una persona puede ser calificada como “alta”, “baja” o de “mediana altura”. La altura

de una persona es un concepto típicamente difuso, al igual que lo puede ser el estado de madurez de una fruta ("maduro", "muy maduro", "poco maduro", "verde") o la temperatura ambiente (hace "calor", "mucho calor",etc.), entre otros.

Formalmente, sea χ un conjunto de objetos no vacío considerado el universo de estudio. Un conjunto difuso es un par (χ, f), donde f es la denominada *función de pertenencia*:

$$f(\chi) \rightarrow [0,1]$$

De este modo, $f(x)$ representa el grado de pertenencia del objeto $x \in \chi$ al conjunto difuso (χ, f). Siguiendo con el ejemplo anterior, una persona de 1,75 metros de altura puede ser clasificada como "alta" con grado de pertenencia 0,6 y a su vez como "baja" con grado de pertenencia 0,4. Es decir, el grado de pertenencia de esa persona al conjunto difuso de las personas "altas" es 0,6, y su grado de pertenencia al conjunto de las personas "bajas" es 0,4; por lo que esa persona pertenece a ambos conjuntos, con un grado de pertenencia asociado a cada uno de ellos.

Las técnicas de agrupamiento clásicas o *crisp* son exclusivas y dividen un conjunto de datos en grupos de objetos similares de forma que cada objeto pertenece exactamente a un único grupo. En las técnicas de agrupamiento difusas, cada objeto puede pertenecer a más de un grupo, con un grado de pertenencia asociado a cada uno de ellos. Es decir, en el clustering difuso los grupos son conjuntos difusos, y el proceso de clustering consistirá en determinar los grados de pertenencia de cada objeto a cada grupo.

1.2.2.1 K-medias difuso.

El algoritmo K-medias difuso, también conocido como *fuzzy c-means* o FCM, es la adaptación del clásico algoritmo de las k medias al universo de los conjuntos difusos.

En el algoritmo clásico, se obtienen los centroides de cada cluster y se determina si cada elemento pertenece a un cluster según su distancia a los centroides. Este proceso se ejecuta de forma iterativa hasta alcanzar un criterio de convergencia determinado. En el FCM se mantiene el concepto de centroide, pero se calcula el grado de pertenencia de cada elemento a cada clusters considerando los clusters como conjuntos difusos.

Sea $X = \{x_i \mid i = 1..N\}$, con $x_i \in \Re^p$, el conjunto de N objetos p-dimensionales que se desean agrupar. Suponemos que X tiene una estructura de k clusters que se describe mediante una partición difusa U, donde k es el número de grupos y $U_{k\times N}$

es la matriz de grados de pertenencia, con u_{ij} representando el grado de pertenencia del objeto j al cluster i. Notamos como $L = \{l_i \mid i = 1..k\}$ la k-tupla de centroides, donde l_i representa el centroide del cluster i. Finalmente, notamos como d la función de distancia entre objetos. El algoritmo K-medias difuso se describe a continuación:

Algoritmo K-medias difuso

INPUT: Conjunto de datos X, número k de clusters de X y cota de error máximo admisible ϵ como criterio de parada.

OUTPUT: Conjunto de centroides L de los k clusters y matriz $U_{k\times N}$ de grados de pertenencia.

1. Construir una partición difusa arbitraria de X, asignando a cada objeto su grado de pertenencia a cada cluster: u_{ij} con $i = 1..k$; $j = 1..N$. Esta partición se realiza de forma que:

$$\sum_{i=1}^{k} u_{ij} = 1;\ \forall j = 1..N$$

2. Repetir:
   a) Calcular los centroides L tal que:

$$l_i = \frac{\sum_{j=1}^{N} u_{ij}^2 \cdot x_j}{\sum_{j=1}^{N} u_{ij}^2}$$

   b) Actualizar los grados de pertenencia U' de la forma:

$$u'_{ij} = \frac{1}{\sum_{z=1}^{k} \frac{d^2(x_j, L_i)}{d^2(x_j, L_z)}}\ i = 1...k;\ j = 1...N$$

3. Hasta que converja, es decir, hasta que la diferencia máxima entre las matrices de grados de pertenencia sea menor que ϵ:

$$\|U - U'\| < \epsilon$$

Respecto a la utilización de las técnicas de clustering en bioinformática, su principal aplicación es el análisis de datos de expresión genética procedente de microarrays. El Capítulo 2 presenta una revisión de los métodos de clustering más importantes aplicados al análisis de microarrays.

1.2.3 Biclustering.

1.2.3.1 Definición y tipos de Bicluster.

El concepto de bicluster fue introducido por Hartigan en 1975 [133]. Sea $X_{N\times p}$, una matriz de N filas y p columnas, y x_{ij} el elemento de la fila i y columna j de $X_{N\times p}$, notamos como $F = \{1..N\}$ el conjunto de filas de $X_{N\times p}$, y $C = \{1..p\}$ el conjunto de columnas. Así, podemos representar $X_{N\times p}$ como X_{FC}. Dados $I \in F$ y $J \in C$, X_{IJ} es la submatriz de filas I y columnas J de $X_{N\times p}$. Notamos x_{IJ} como la media de la matriz X_{IJ}, x_{iJ} como la media de la fila i para las columnas de J y x_{Ij} como la media de la columna j para las filas de I.

Un bicluster B se define como una submatriz $B = X_{IJ}$ de $X_{N\times p}$, con valores correlados de acuerdo a un determinado criterio. Existen en la literatura algoritmos de biclustering que consideran distintos criterios [210] (ver Figura 1.10):

- Con valores constantes: Un bicluster perfecto será una submatriz tal que todos los valores son constantes: $x_{ij} = \lambda$
- Con valores constantes en filas o en columnas: En los biclusteres con valores constantes en filas, cada entrada x_{ij} de la matriz es igual a una constante de fondo más un valor dependiente de la fila i: $x_{ij} = \lambda + \phi_i$. Alternativamente, en biclusteres con valores constantes en columnas, cada entrada x_{ij} es igual a una constante de fondo más un valor dependiente de la columna j: $x_{ij} = \lambda + \gamma_j$.
- Con valores coherentes: Exiten dos modelos, el aditivo y el multiplicativo y ambos son equivalentes. En el modelo aditivo (multiplicativo) cada entrada (i, j) de la matriz se puede considerar como suma (multiplicación) de una constante de fondo λ, una constante ϕ_i que depende de la fila i y una constante γ_j que depende de la columna j.

  $$x_{ij} = \lambda + \phi_i + \gamma_j \, (modelo\, aditivo)$$

  $$x_{ij} = \lambda * \phi_i * \gamma_j \, (modelo\, multiplicativo)$$

- Con evolución coherente: Se trata de biclusteres que presentan una evolución coherente en sus filas, columnas o filas y columnas simultáneamente sin importar los valores exactos.

El critero de valores coherentes engloba a los biclusters con valores constantes por filas o columnas, que a su vez engloban a los biclusters con valores constantes. El

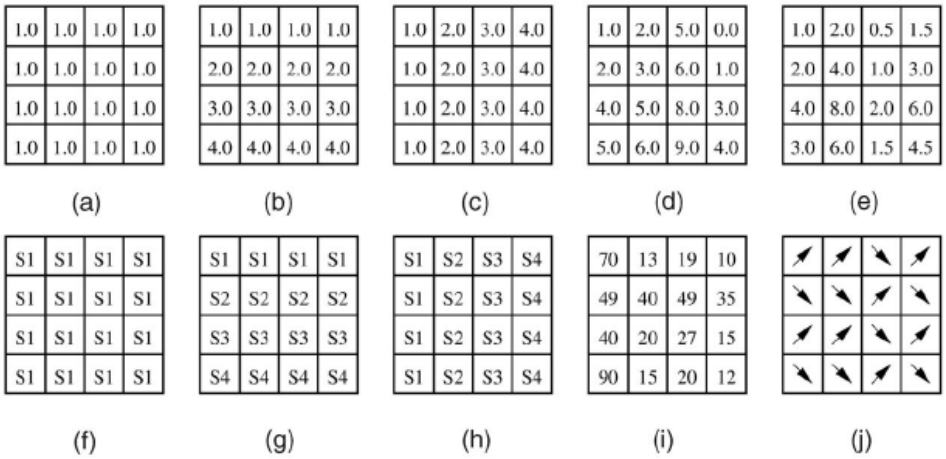


Figura 1.10: *Tipos de bicluster según el criterio de correlación de valores. (a) bicluster constante (b) bicluster con valores constantes por filas (c) bicluster con valores constantes por columnas (d) bicluster con valores coherentes (modelo aditivo) (e) bicluster con valores coherentes (modelo multiplicativo) (f) bicluster con evolución coherente (g) bicluster con evolución coherente en filas (h,i) biclusters con evolución coherente en columnas (j) bicluster con cambios de signo coherentes en filas y columnas. Tomado de [210].*

criterio de evolución coherente es un caso especial de bicluster, en el que las entradas son consideradas como etiquetas en lugar de valores numéricos, y se busca una evolución coherente de dichas etiquetas en el bicluster.

El bicluster con valores coherentes es el más potente y el que mejor se adapta a la estructura de las matrices de expresión genética: el modelo recoge genes expresados en diferentes cantidades a lo largo de las condiciones, pero siguiendo una evolución coherente, esto es, sobre-expresados o sub-expresados bajo las mismas condiciones. Lo mismo puede decirse de las condiciones.

En cualquier caso, encontrar biclusters de tamaño máximo en una matriz es un problema NP-completo [245], por lo que la mayor parte de los algoritmos existentes en la literatura utilizan aproximaciones heurísticas [210].

Otro segundo criterio permite clasificar los métodos de biclustering según la estructura y solapamiento de los biclusters obtenidos dentro de la matriz de datos. Podemos distinguir los siguientes tipos [210] (ver Figura 1.11):

- Un único bicluster en la matriz de datos. Algunos algoritmos asumen la existencia de un único bicluster en la matriz de datos y, partiendo de esta hipótesis, tratan de identificarlo. Sin embargo, la mayoría de los algoritmos asumen la existencia de un número determinado de biclusters k.

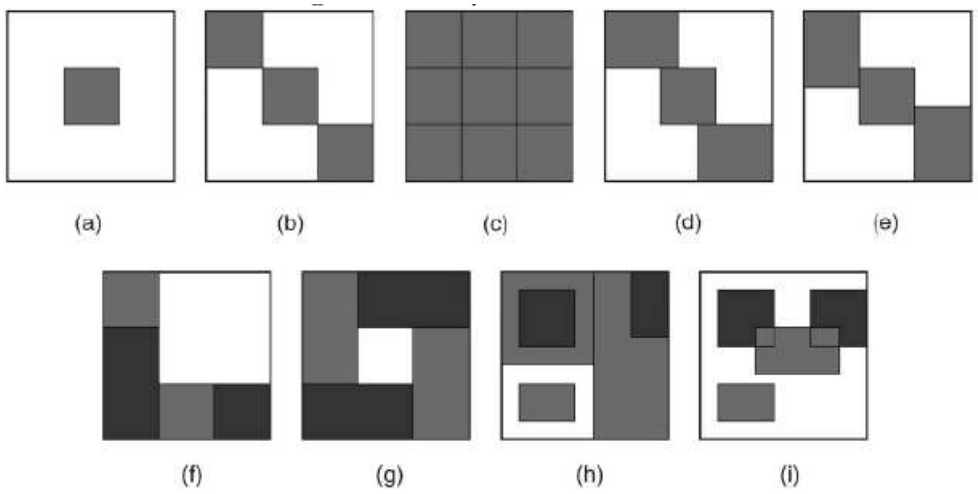


Figura 1.11: *Tipos de bicluster según su estructura en la matriz y las restricciones de solapamiento. (a) un único bicluster (b) biclusters que no comparten filas ni columnas (c) estructura de tablero de ajedrez (d) biclusters que no comparten filas (e) biclusters que no comparten columnas (f) biclusters no solapados con estructura de árbol (g) biclusters no solapados no exclusivos (h) biclusters solapados con estructura jerárquica (i) biclusters solapados arbitrariamente posicionados. Tomado de [210].*

- Biclusters que no comparten filas ni columnas. Una fila o una columna de la matriz de datos estará asociada, como máximo, a un único bicluster.
- Biclusters que no comparten filas (columnas). Una fila (columna) de la matriz de datos estará asociada, como máximo, a un único bicluster.
- Biclusters en estructura de tablero de ajedrez. Se asume la existencia de $k \times n$ bicluster exclusivos y no solapados en los datos. Cada fila pertenece a n biclusters mientras que cada columna pertenece a k biclusters.
- Biclusters no solapados con estructura de árbol. Se asume la existencia de una estructura en árbol para los biclusters de la matriz de datos. Es decir, la región de la matriz se descompone de forma jerárquica y recursiva en áreas según un determinado criterio. Los nodos hoja de esta estructura en árbol constituyen los biclusters.
- Biclusters solapados con estructura jerárquica. Este tipo de bicluster es similar al anterior, pero se permite que los biclusters se solapen.
- Biclusters no solapados no exclusivos. Una misma fila o columna puede pertenecer a varios biclusters, pero no puede existir solapamiento entre los mismos.

Es decir, si la fila i y la columna j pertenecen a un bicluster, i y j no pueden pertenecer a otro bicluster distinto.

- Biclusters solapados arbitrariamente posicionados. Se trata de la estructura más general, que no impone restricciones sobre la posición de los biclusters en la matriz o el solapamiento entre los mismos.

Respecto a la utilización de las técnicas de biclustering en bioinformática, su principal aplicación ha sido el análisis de datos de expresión genética procedente de microarrays. Los Capítulos 2 y 3 incluyen sendas secciones que revisan métodos de biclustering aplicados al análisis de microarrays.

1.2.4 Clasificación.

Dado un conjunto de objetos $D = \{x_i \mid i = 1..N\}$, descritos por las características $x_i = \{x_{ij} \mid j = 1..p\}$, que pueden ser asociados a una clase de un conjunto de clases posibles $y = \{y_z \mid z = 1..k\}$, la clasificación se define como la tarea de asignar a cada objeto su clase asociada. Para determinar la clase correcta, un clasificador necesita, por tanto, describir una función discreta, un mapeo del espacio de características de un objeto al espacio de clases: $f : x_i \rightarrow y$. Para aprender esta función f, los clasificadores hacen uso de un conjunto de entrenamiento D_T para los que se conoce su clase asignada. La clasificación es, por tanto, un tipo de método de aprendizaje supervisado.

Existen numerosos problemas de clasificación en el campo de la bioinformática [209]. Algunos ejemplos se comentan a continuación:

- Clasificación de tipos de cáncer en microarrays. Los métodos de clasificación han sido ampliamente utilizados para identificar biomarcadores característicos de distintos fenotipos tumorales, incluyendo cancer de colon, próstata, pecho, piel o linfomas entre otros. Algunos trabajos pioneros en esta línea son los de Golub *et al.* [125] o West *et al.* [329]. Posteriormente, diversos sistemas han sido puestos a disposición de la comunidad para la clasificación de muestras en microarrays, como Macbeth [250], Prophet [220] o MAMA [30].
- Clasificación de tipos de cáncer con epigenética. Además de causas genéticas, el cancer puede considerarse una enfermedad epigenética. La regulación epigenética implica alteraciones en la estructura de la cromatina y la metilación de regiones promotoras, por lo que medidas epigenéticas, como patrones de metilación de ADN, pueden también emplearse para la clasificación de tipos de cáncer [350].

- Clasificación en proteómica. Técnicas relacionadas con la proteómica han sido también empleadas para caracterizar distintos objetos y clasificarlos.
  - La espectrometría de masas es una técnica que permite determinar la composición de una muestra. Algunos tipos de tumores alteran las concentraciones de determinadas moléculas en sangre, por lo que la espectrometría de masas puede ayudar a discriminar entre individuos con distintos fenotipos tumorales. Esta técnica ha sido empleada para la caracterización de cancer de próstata, ovario, pecho, páncreas, riñón, colon e hígado, entre otros [299, 91].
  - Clasificación de secuencias de aminoácidos en familias funcionales y estructurales. Habitualmente resulta sencillo obtener la secuencia de aminoácidos de una proteina, pero muy complejo identificar su estructura. Numerosos trabajos han abordado la clasificación de secuencias de proteínas en familias y super-familias definidas por relaciones funcionales y estructurales [200, 330, 163].
  - La determinación de la localización celular de las proteinas resulta de gran interés para la identificación de las funciones bioquímicas de las mismas. Los métodos computacionales de predicción de localización celular hacen uso de características de la secuencia de aminoácidos de las proteínas para asignarles una localización celular, en base a su similitud con un conjunto de proteínas para las que se conoce su localización [339, 292].

Los distintos métodos de clasificación utilizan diversas representaciones para la función de mapeo f. Algunos de los paradigmas de clasificación más comunes son los árboles y reglas de decisión, clasificadores Bayesianos, clasificadores basados en los vecinos más cercanos, funciones discriminantes, regresión logística, máquinas de vector soporte (*support vector machines*, SVM) o redes neuronales artificiales, entre otros [185, 190]. En esta sección presentamos brevemente algunos de los paradigmas que resultan de interés para el contenido de esta memoria: los clasificadores bayesianos y la regresión logística.

1.2.4.1 Clasificación bayesiana.

Dado un nuevo objeto, un clasificador bayesiano predice la clase asociada a dicho objeto mediante el cálculo de las probabilidades condicionadas para todas las clases posibles. Teniendo en cuenta la siguiente notación:

- $P(C_k)$: probabilidad a priori de las clases C_k, $k = 1, ..., m_0$
- $V =< v_1, ..., v_a >$: vector de características que describen el objeto V.
- $d(V)$: mapeo del clasificador del objeto V a una clase.
- $t(V)$: clase correcta del objeto V.
- $P(V)$: probabilidad *a priori* del objeto V.
- $P(V \mid C_k)$: probabilidad condicionada de V dada la clase C_k.

Un clasificador bayesiano $d_B(V)$ se define como el clasificador que minimiza la probabilidad de error de clasificación sobre todos los clasificadores posibles:

$$\forall d(\cdot) : P(d_B(V) \neq t(V)) \leq P(d(V) \neq (t(V)))$$

La tasa de acierto promedio (*accuracy*) de un clasificador bayesiano $d_B(V)$ viene dada por:

$$Acc_B = P(d_B(V) = t(V))$$

Por definición, el clasificador bayesiano cumple:

$$d_B(V) = argmax_{C_k} P(C_k \mid V) = argmax_{C_k} P(V \mid C_k)P(C_k)$$

por lo que su tasa de acierto promedio es:

$$Acc_B = \sum_V max_{k \in \{1..m_0\}} \{P(C_k \mid V)P(V)\}$$

Un clasificador bayesiano exacto minimiza la tasa de error esperado, por lo que resulta el mejor clasificador posible. Sin embargo, dado que las probabilidades a priori y condicionales son desconocidas, éstas deben ser estimadas del conjunto de datos de entrenamiento, lo que resulta en una pérdida de acierto importante en la práctica. Es por ello que, en la práctica, es habitual la asunción de ciertas propiedades sobre los datos para reducir la complejidad del modelo y simplificar el proceso de estimación de probabilidades condicionales y a priori.

Un clasificador bayesiano *Naive* (del inglés *Naive Bayesian*, traducido directamente como Bayesiano *ingenuo* o *simple*), es un clasificador bayesiano que asume la independencia del objeto V dada la clase C_k. Según la regla de Bayes:

$$P(C_k \mid V) = P(C_k)\frac{P(V \mid C_k)}{P(V)}$$

asumiendo la independencia condicional de V dada la clase C_k:

$$P(V \mid C_k) = P(v_1 \wedge ... \wedge v_a \mid C_k) = \prod_{i=1}^{a} P(v_i \mid C_k)$$

aplicando la regla de Bayes se obtiene:

$$P(C_k \mid V) = \frac{P(C_k)}{P(V)} \prod_{i=1}^{a} P(v_i \mid C_k)$$

$$P(v_i \mid C_k) = P(v_i) \frac{P(C_k \mid v_i)}{P(C_k)}$$

de donde se obtiene:

$$P(C_k \mid V) = P(C_k) \frac{\prod_{i=1}^{a} P(v_i)}{P(V)} \prod_{i=1}^{a} \frac{P(C_k \mid v_i)}{P(C_k)}$$

El factor $\frac{\prod_{i=1}^{a} P(v_i)}{P(V)}$ es independiente de la clase, por lo que omitiéndolo se obtiene la expresión:

$$P(C_k \mid V) = P(C_k) \prod_{i=1}^{a} \frac{P(C_k \mid v_i)}{P(C_k)}$$

La tarea de aprendizaje del algoritmo consiste en utilizar los objetos del conjunto de entrenamiento para aproximar las probabilidades (tanto condicionadas como no condicionadas) de la parte derecha de esta ecuación. Por lo tanto, el conocimiento almacenado en un clasificador Naive Bayes se representa como el conjunto de las probabilidades a priori de cada clase $P(C_k), k = 1..m_0$ y la probabilidad condicionada de las clases $P(C_k \mid v_i), k = 1..m_0$ dadas las características v_i de los objetos V.

1.2.4.2 Clasificación por regresión logística.

La regresión logística es un método de clasificación basado en la predicción de la probabilidad de la clase en función de las características de los objetos. La regresión logística es un caso especial de modelo lineal generalizado (MLG). Los MLG asumen que la variable dependiente (en este caso, la clase c) está generada por una función expresada respecto a las variables independientes (en este caso, las características del objeto):

$$\widehat{f(c)} = \beta_0 + \sum_{i=1}^{a} \beta_i v_i \tag{1.1}$$

donde $\widehat{f(c)}$ es la denominada *función de enlace*. Si $f(c) = c$ la expresión 1.1 describe un modelo de regresión lineal. Otras funciones habituales para $f(c)$ son $f(c) = 1/c$, $f(c) = log(c)$ y $f(c) = log(c/(1-c))$ (denominado función *logit*). Para problemas de clasificación en dos clases, los MLG pueden ser fácilmente adoptados haciendo $c = P(C_1)$, con C_1 representando una de las dos clases. Si además utilizamos la función *logit*, obtenemos la expresión que define la regresión logística:

$$log\left(\frac{P(C_1)}{1-P(C_1)}\right) = \beta_0 + \sum_{i=1}^{a} \beta_i v_i$$

Eliminando logaritmos obtenemos la expresión:

$$P(C_1) = \frac{1}{1 + exp - (\beta_0 + \sum_{i=1}^{a} \beta_i v_i)}$$

Los coeficientes $\beta = \beta_i \mid i = 0, ..., a$ son los parámetros del modelo de regresión y se estiman a partir del conjunto de entrenamiento, minimizando una función de error determinada (en el caso de la clasificación binaria, la función de error es la entropía cruzada [185]). Este modelo básico es extendido en el capítulo 5, en el que se propone un modelo de regresión logística *multinomial* (para la predicción de más de dos clases) para la clasificación de historiales clínicos en distintos diagnósticos.

1.2.4.3 Estimación del error de clasificación.

Una cuestión importante relacionada con el diseño de clasificadores es la estimación de la tasa de error cuando se utiliza un clasificador para la predicción de clases de nuevos objetos. Esta introducción no pretende ser una revisión completa de métodos (disponible en [190, 37]), sino una mera introducción a las técnicas y medidas más importantes que serán utilizadas en otras secciones de esta memoria.

En el contexto de la clasificación, se emplean los términos *true positive (tp)*, *true negative (tn)*, *false positive (fp)* y *false negative (fn)* para describir si la clase asignada a un objeto por un clasificador se corresponde con su clase real. Esto queda ilustrado en la figura 1.12.

En base a estos conceptos, se definen las medidas *precision*, *recall* y *f-measure* como:

$$precision = \frac{tp}{tp + fp}$$

$$recall = \frac{tp}{tp + fn}$$

		correct result / classification	
		E1	**E2**
obtained result / classification	**E1**	**tp** (true positive)	**fp** (false positive)
	E2	**fn** (false negative)	**tn** (true negative)

Figura 1.12: *Definición de true positive, true negative, false positive y false negative en un problema de clasificación. Imagen tomada de Wikipedia: http://en.wikipedia.org/wiki/Precision_and_recall*

$$f - measure = \frac{2 \times precision \times recall}{precision + recall}$$

Respecto a los métodos de estimación del error de un clasificador, el método más sencillo para efectuar esta estimación sin hacer uso de datos de test, es calcular el error en el propio conjunto de entrenamiento. En la validación cruzada *k-fold*, el conjunto de entrenamiento D_T es particionado en k subconjuntos: $D_T^{(i)} \mid i = 1, ..., k$, de modo que cada uno de los subconjuntos de objetos es excluido del conjunto de entrenamiento y utilizado como conjunto de test en cada una de las k ejecuciones. La estimación del error del clasificador $Error_v ck$ se calcula como el error promedio utilizando cada uno de los k subconjuntos de test:

$$Error_v ck = \frac{1}{N} \sum_{i=1}^{k} \sum_{j=1}^{N/k} \mid y_j^{(i)} - f(D_T \setminus D_T^{(i)}, x_j^{(i)}) \mid$$

donde $(x_j^{(i)}, y_j^{(i)})$ es el j-esimo objeto del subconjunto $D_T^{(i)}$, $D_T \setminus D_T^{(i)}$ representa el conjunto de entrenamiento excluyendo los objetos del conjunto $D_T^{(i)}$, $f(D_T^{\cdot}, x)$ es la clase predicha para el objeto x utilizando el conjunto de entrenamiento $D_T^{\cdot}$ y $\mid \cdot \mid$ representa la función que mide el error de clasificación de un objeto.

En la validación cruzada *leave-one-out* (o deja-uno-fuera), se excluye un único objeto del conjunto de entrenamiento en cada ejecución, lo que equivale a una validación cruzada en N subconjuntos.

1.2.4.4 Selección de características.

Otro aspecto importante a considerar en el diseño de clasificadores, es si las N características que describen a los objetos son útiles para aprender la clase asociada a los mismos. Las técnicas de selección de características (*Feature Subset Selection*, FSS) tratan de resolver este problema. En función del grado de integración con el clasificador, existen tres tipos de métodos de selección de características:

- Métodos de tipo filtro (*filter approaches*): separan la selección de características de la construcción del clasificador, de modo que la selección de características es un paso previo a la construcción del clasificador.
- Métodos de tipo envolvente (*wrapper approaches*): evalúan la bondad del clasificador con un cierto subconjunto de características, continuando la búsqueda hasta que se obtiene el subconjunto que optimiza la bondad de clasificación.
- Métodos incrustados (*embedded approaches*): la selección de características forma parte de la construcción del clasificador.

La reducción de dimensionalidad que producen los métodos de FSS conlleva varias ventajas respecto al sistema de clasificación, como la reducción del coste de adquisición y almacenamiento de datos, la simplificación del modelo representado por el clasificador, la reducción del tiempo de aprendizaje y el aumento de la bondad de clasificación.

La selección de características puede verse como un problema de búsqueda, en el que una solución del espacio de búsqueda representa un subconjunto de características. La evaluación exhaustiva de todas las soluciones posibles en el espacio de características es habitualmente inviable, tratándose de un problema NP-duro: para N características, el espacio total de posibles soluciones es 2^N. En problemas de gran dimensionalidad, los métodos heurísticos estocásticos de búsqueda constituyen una solución efectiva para la selección de características, habiendo sido aplicados a este problema fundamentalmente algoritmos genéticos [296] y algoritmos de estimación de distribuciones de probabilidad (EDA) [268]. Para una revisión completa de métodos de FSS y su aplicación en biomedicina, consultar [269, 209].

1.3 Sistemas de extracción de información de textos biomédicos.

1.3.1 Análisis automático de la literatura biomédica.

Los avances actuales en las técnicas biológicas de producción masiva de datos están acompañados por un enorme incremento del volumen de publicaciones científicas que difunden los descubrimientos y conclusiones extraídos del análisis de este tipo de datos [282]. De este modo, la literatura biomédica crece a una tasa exponencial [149]. En los últimos 20 años, el tamaño total de MEDLINE (*Medical Literature Analysis and Retrieval System Online*), la mayor base de datos bibliográfica mundial, mantenida por *U.S. National Library of Medicine*(NLM), ha crecido a una tasa anual cercana al 4,2 % y el número de nuevas publicaciones al 3,1 % (ver Figura 1.13). En la actualidad MEDLINE contiene más de 19 millones de publicaciones, de las cuales más de 3.5 millones fueron publicadas en los últimos 5 años [183].

El enorme volumen de información disponible plantea retos importantes a la comunidad científica. Por ejemplo, una búsqueda directa del término *autism* sobre PubMed (http://www.ncbi.nlm.nih.gov/pubmed), el motor de búsqueda de MEDLINE, devuelve más de 14,000 artículos relacionados; la búsqueda del gen *p53* devuelve más de 50,000 artículos[2]. Este ejemplo es sólo una muestra del enorme volumen de información al que se enfrenta un investigador en un determinado campo.

Una parte del conocimiento disponible en la literatura científica se encuentra almacenado de forma estructurada en bases de datos y ontologías especializadas. Estos recursos están diseñados para proporcionar un acceso eficiente a los datos y facilitar su manejo y análisis. La mayor parte de estos recursos son cargados de contenido extraído de la literatura por medio de anotadores humanos. De este modo, aunque las bases de datos de propósito general, como Uniprot (que contiene casi 600,000 referencias a la literatura[3]), son de extraordinario valor, estos recursos sólo recogen una pequeña fracción de la información publicada en la literatura. Incluso si se emplearan suficientes anotadores humanos para mantener cualquiera de estos recursos actualizado respecto al volumen creciente de artículos publicados, los investigadores aún necesitarían utilizar técnicas automáticas para extraer conocimiento adicional de relevancia, como la evidencia textual que soporta una anotación o detalles sobre la misma [187, 43].

[2]Búsquedas efectuadas sobre MEDLINE en Noviembre de 2009

[3]Datos consultados en `http://www.uniprot.org` en diciembre de 2009.

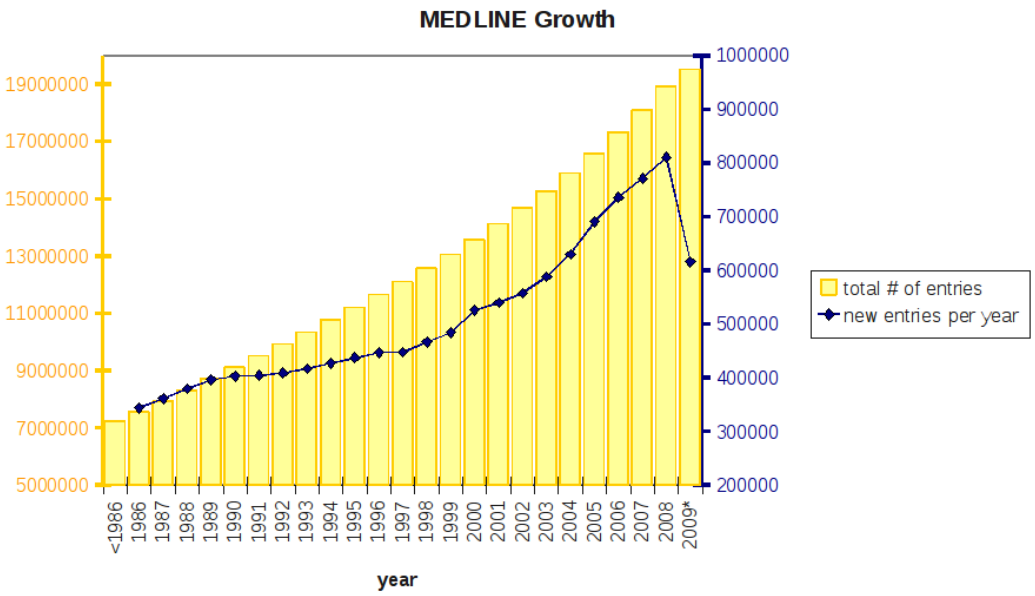


Figura 1.13: *Evolución del volumen de artículos en MEDLINE.*

De este modo, la utilización de técnicas automáticas de minería de textos (o *text-mining*) resulta fundamental para extraer automáticamente el conocimiento de la literatura científica y *poblar* las bases de datos biológicas y bio-ontologías. Formalmente, el objetivo principal de las técnicas de *text-mining* es recuperar el conocimiento oculto en el texto y presentarlo al usuario de forma concisa.

Existen distintas definiciones de *text-mining* ([153], [136]). En sentido estricto, un sistema de *text-mining* debe extraer conocimiento que no está explícitamente constatado en el texto analizado. En un sentido más amplio, cualquier sistema que extrae conocimiento de forma automática a partir de texto puede considerarse como sistema de *text-mining* [351]. Hearst [137] caracteriza el *text-mining* como el proceso de descubrimiento y extracción de conocimiento a partir de datos sin estructura, en contraposición con el *data-mining*, que descubre conocimiento sobre datos estructurados. Según esta visión, el *text-mining* comprende tres actividades principales: la recuperación de información (o *information retrieval*, IR), para recuperar textos de relevancia; la extracción de información (*information extraction*, IE), para identificar y extraer cierto tipo de información de los textos de interés; y el *data-mining*, para encontrar relaciones entre piezas de información extraídas de los textos de interés.

La minería de textos se aplica en numerosas áreas de la bioinformática, como el

descubrimiento de relaciones funcionales entre genes, la identificación de anotaciones funcionales, detección de relaciones entre entidades biomédicas de interés (como asociaciones proteína-proteína, gen-enfermedad o gen-fenotipo, proteína-localización celular, etc.), el desarrollo de fármacos o la interpretación y validación de resultados obtenidos con técnicas experimentales, entre otras [28, 57, 19].

La mayoría de técnicas de *text mining* se apoyan de una forma u otra en métodos y herramientas de Procesamiento de Lenguaje Natural (*Natural Language Processing*, NLP). En el procesamiento del lenguaje natural podemos distinguir distintos niveles descriptivos y de análisis:

- Nivel léxico, centrado en la palabra (o token). En este nivel pueden incluirse los problemas relacionados con la identificación y delimitación de las distintas palabras que componen una cadena de texto (*tokenización*), o la identificación de variantes de palabras creadas por adición de prefijos, sufijos o derivaciones (análisis morfológico).

- Nivel sintáctico, centrado en la organización de palabras en cláusulas para componer las oraciones. En este nivel se encuadran problemas como la identificación de la etiqueta POS (*Part-of-Speech*, parte de la oración) de cada palabra (por ejemplo, artículo, sustantivo, adjetivo, verbo, preposición, adverbio, etc.); el *chunking* o identificación de grupos de palabras que presentan una misma función gramatical en una oración (habitualmente el interés recae en frases nominales o *Noun-phrases*); y el *parsing* o análisis sintáctico de las oraciones para determinar su estructura gramatical.

- Nivel semántico, centrado en el significado o *mensaje* del texto. En este nivel se encuadran los problemas relacionados con el mapeo de términos del texto en conceptos y relaciones estándares y la representación semántica del mensaje transmitido en el texto.

El lenguaje natural está además sujeto a peculiaridades asociadas al contexto o dominio de discurso, en este caso, la biomedicina. Para la comunicación de información en biomedicina, los expertos hacen uso de una extensa terminología muy específica (nombres de genes, proteínas, organismos, procesos biológicos, etc.) y de convenciones tipográficas y ortográficas (por ejemplo, existe un uso extendido de acronismos para identificar entidades biomédicas de interés y términos de uso frecuente en el campo). El trabajo de Bernardi *et al.* [46] estima que más del 12 % de las palabras de los textos bioquímicos pertenecen a la terminología específica de esta disciplina. La

1	Neurofibromatosis type 2 (NF2) is often not recognized as a distinct entity from peripheral neurofibromatosis. NF2 is a predominantly intracranial condition whose hallmark is bilateral vestibular schwannomas. NF2 results from a mutation in the gene named merlin, located on chromosome 22. [PMID: 16140520]
2	We report a positive association between autism and two HRAS markers. [PMID: 7485261]
3	The action of SCPA enzymatically inhibits the chemotactic activity of C5a by cleaving its neutrophil binding site. [PMID: 12964111]
4	Three promoter, one intronic, and one 3' UTR single nucleotide polymorphisms (SNPs) in the APOE gene (-491a/t, -427c/t, -219g/t, 113c/g, and 5361c/t) as well as the APOE functional polymorphism (E2, E3, E4) were examined and failed to reveal significant evidence that autism is associated with APOE. [PMID: 14755445]
5	While stimulation of the D2 receptor increased branching and extension of neurites, stimulation of the D1 receptor reduced neurite outgrowth, suggesting that hormones and neurotransmitters may be capable of controlling the development of specific types of neurones.

Cuadro 1.3: *Extractos de textos biomédicos.*

Tabla 1.3 muestra algunos extractos de texto obtenidos de artículos científicos reales, en los que pueden observarse las características mencionadas.

Existen numerosos campos de investigación relacionados con el *text-mining* de textos biomédicos:

- Detección de ocurrencias en el texto de entidades biológicas (*Named Entity Recognition*, NER), por ejemplo, nombres de genes, proteinas, enfermedades, organismos o términos de bio-ontologías, entre otras.
- Normalización de las entidades biológicas identificadas en el texto. Esto equivale a establecer un mapeo entre las ocurrencias de las entidades detectadas, por ejemplo genes o proteínas, y sus correspondientes entradas en recursos como UniGene o Uniprot, en este caso.
- Extracción de relaciones entre entidades biológicas a partir del texto. Asociado con este campo también existe un creciente interés por el descubrimiento de nuevos hechos biológicos no constatados directamente en la literatura, pero que pueden ser deducidos de la combinación de distintas relaciones extraídas de distintas publicaciones.
- Resumen de publicaciones (*Document Summarization*) para presentar sólo los hechos más importantes constatados en un artículo.
- Procesamiento de información no textual (imágenes, tablas, gráficas, etc).
- Clasificación, recuperación y ranking de textos según un tema o criterio de interés.

1	Genes: Tp53, merlin, HRAS, agaR.
2	Proteins: p53, “galactosidase, alpha(GLA)”, “human T-cell leukaemia lymphotropic virus type 1 Tax protein’.”
3	Drugs: herbimycin, lantus
4	Diseases: autism, Autism Spectrum Disorder, ASD
5	Chemicals: 5’-(N-ethylcarboxamido)adenosine (NECA)

Cuadro 1.4: *Ejemplos de NEs de conceptos biomédicos.*

El esfuerzo comunitario más conocido para la evaluación de sistemas de *text-mining* en el dominio de la biomedicina es el proyecto BioCreative (*Critical Assessment of Information Extraction systems in Biology*) [141, 186]. El objetivo de este proyecto es proponer tareas de extracción de información de la literatura que lleven al desarrollo de sistemas que puedan ser utilizados por científicos de campo, investigadores y anotadores de bases de datos especializadas. Estas tareas incluyen la detección y normalización de nombres de genes y proteínas en extractos de texto extraídos de abstracts de PubMed; la detección de asociaciones de genes y proteínas con términos de *Gene Ontology*; la recuperación de artículos con interacciones entre genes/proteínas y la identificación y normalización de dichas interacciones y sus agentes.

1.3.1.1 Detección de entidades de interés en textos biológicos.

La identificación de ocurrencias de entidades biológicas de interés (NER) en textos biomédicos supone un paso previo necesario para la aplicación de técnicas de minería de textos. En el campo de la biomedicina, definimos una *named entity* o NE como un segmento de texto que denota un objeto o conjunto de objetos específicos, como genes, proteínas, componentes químicos, fármacos, enfermedades, etc. La Tabla 1.4 muestra ejemplos de distintos tipos de NEs. En [28] se define el objetivo principal de las técnicas de NER como la asociación de un segmento de texto en lenguaje natural con su entidad correspondiente en el mundo real.

La detección de entidades de interés está íntimamente relacionada con el diseño de terminologías [291] y ontologías para la anotación de textos y datos experimentales [280], en procesos que habitualmente requieren un gran esfuerzo comunitario [150, 229]. De este modo, existe una amplia variedad de recursos léxicos y terminológicos disponibles para asistir a las técnicas de NER en biomedicina. Algunos de estos recursos son brevemente descritos a continuación:

- UMLS (*Unified Medical Language System*). El proyecto UMLS tiene como objetivo la distribución de bases de datos y *software* para el desarrollo de sistemas

automáticos que creen, manipulen, recuperen, integren, y/o agregen información biomédica. Algunos de sus recursos más utilizados en el ámbito del NLP en biomedicina son el metatesauro (`http://www.nlm.nih.gov/research/umls/knowledge_sources/metathesaurus/index.html`), una base de datos que contiene información sobre conceptos biomédicos, sus sinónimos y otros conceptos relacionados; o el léxico SPECIALIST (`http://lexsrv3.nlm.nih.gov/SPECIALIST/index.html`), un léxico de términos biomédicos que proporciona información sintáctica (etiqueta POS), morfológica (raíz del término y prefijos/sufijos) y ortográfica (variantes ortográficas).

- HUGO (*Human Genome Organization*). Para cada gen conocido del genoma humano, el HUGO *Gene Nomenclature Committee* (HGNC) aprueba un nombre y símbolo (nombre corto) para dicho gen, manteniendo una base de datos de todos los genes conocidos. El HGNC garantiza que cada símbolo es único, y que a cada gen se le asigna un único símbolo. Los esfuerzos de la HUGO están orientados al establecimiento de una nomenclatura universal para los genes del genoma humano, de modo que cuando un símbolo aparezca en un artículo, no exista ambiguedad sobre a qué gen se refiere. Actualmente la base de datos del HGNC contiene más de 24,000 simbolos incluyendo genes que codifican proteinas, pseudogenes, RNAs, fenotipos y características genéticas.
- GO (*Gene Ontology*). GO [32] es una ontología desarrollada por el *Gene Ontology Consortium*, para la anotación de productos de genes. GO está organizado en tres ontologías: función molecular, proceso biológico y componente celular. Cada ontología es un grafo dirigido acíclico en el que los nodos son términos y los arcos representan relaciones *is-a* o *part-of*. Un término será tanto más general cuanto más cercano esté situado respecto al término raíz de la ontología, y tanto más específico cuanto más cercano esté de los términos hoja. Los genes y productos de genes se anotan en uno o varios términos de GO al nivel más específico posible. Si un producto de gen se anota en un término, también queda indirectamente anotado en todos sus términos ancestros. La anotación no sólo incluye la fuente sino también la evidencia que la soporta. Tanto los nombres de los términos como las relaciones incluidas en GO pueden ser utilizadas en las aplicaciones de text mining, ya que los nombres de las funciones moleculares, procesos biológicos, y componentes celulares son frecuentemente utilizados en la literatura biomédica [217].
- MeSH (*Medical Subject Headings*). MeSH [54] es un tesauro creado por la *Na-*

tional Library of Medicine utilizado para indexar, catalogar y buscar información en la literatura biomédica. MeSH contiene 25,186 descriptores (palabras clave) en su versión de 2009. Los descriptores se organizan en categorías jerárquicas, incluyendo las categorías "Anatomy", "Organisms", "Diseases", "Chemical and Drugs", "Information Science", "Humanities", "Heath Care", entre otros. Al igual que GO, MeSH puede ser utilizado en una amplia variedad de aplicaciones de *text-mining* por la gran cantidad de términos y relaciones que proporciona.

- BioLexicon. BioLexicon [273] es un recurso linguístico del proyecto BOOTStrep (`http://www.bootstrep.org`) que contiene terminología biomédica, incluyendo verbos específicos del campo con sus variaciones ortográficas y formas derivadas; términos biomédicos (nombres de genes, proteínas, etc.) con sus variantes y sinónimos; y léxico del lenguaje general.
- Léxicos para genes y enfermedades:
  - Recursos del NCBI: `http://www.ncbi.nlm.nih.gov/disease/`
  - Disease Database: `http://www.diseasedatabase.com/`
  - BioThesaurus: `http://pir.georgetown.edu/pirwww/iprolink/biothesaurus.shtml`
- Léxicos para nombres de fármacos:
  - MedMaster: `http://www.ashp.org`
  - USP DI: `http://www.usp.org/`
  - DrugBank: `http://redpoll.pharmacy.ualberta.ca/drugbank/`
  - RXList: `http://www.rxlist.com`
  - DRUGLib: `http://www.druglib.com`
- Léxicos para proteínas:
  - UniProt: `http://www.expasy.org/sprot/`
  - IPI: `http://www.ensembl.org/IPI/`

A pesar de la existencia de esta gran variedad de recursos léxicos y terminológicos, el reconocimiento de entidades en biomedicina resulta especialmente dificultoso debido a diversas razones. En primer lugar, existe una gran ambiguedad en el lenguaje natural utilizado en la literatura biomédica, por ejemplo, entre nombres de genes y de las proteinas que codifican (el gen *tp53* y la proteina *p53*). En segundo lugar, existen numerosas variaciones de la ortografía de los nombres de entidades,

un uso inconsistente de acrónimos y nomenclatura (a pesar de esfuerzos como los del HGNC), nombres encadenados y anidados, utilización de sinónimos y el descubrimiento de nuevos genes de forma constante y la asignación de nuevos nombres para los mismos [28].

Dada la importancia de la detección de entidades de interés para la aplicación de técnicas de *text-mining* sobre textos biomédicos, el NER ha sido uno de los campos que más atención ha atraido dentro del procesamiento de textos biomédicos. Mientras que los primeros resultados obtenidos por sistemas basados en reglas fueron prometedores [112], estudios posteriores y más amplios revelaron que la NER en textos biomédicos resultaba más compleja de lo esperado. De este modo, el reconocimiento de entidades biomédicas ha sido el centro de atención de numerosos congresos y evaluaciones competitivas, como JNLPBA [175], y las evaluaciones BioCreative [141], BioCreative II [331] y el recientemente celebrado BioCreative II.5 [18].

La evaluación competitiva JNLPBA utiliza una versión del corpus GENIA (descrito en la Sección 4.2) restringido a cinco tipos de entidades de interés: proteinas, DNA, RNA, lineas celularles y tipos celulares. La evaluación competitiva BioCreative en su tarea 1A (búsqueda de menciones de genes) trata la identificación de un tipo de entidad (gen/proteína). El corpus de BioCreative II para la identificación de menciones de genes (*gene mention*, GM) extiende el corpus de la primera edición con un 50 % más de datos de entrenamiento.

La Tabla 1.5 muestra los mejores resultados obtenidos en estas evaluaciones competitivas [254]. La mejora entre los resultados obtenidos en las dos evaluaciones de BioCreative resulta prometedora, incluso considerando el mayor tamaño del conjunto de entrenamiento para BioCreative II. Los organizadores del evento mostraron, además, que una combinación de la salida de todos los sistemas presentados podría alcanzar el 90,7 de *F-measure*. A pesar de ello, las tasas de error son considerablemente superiores que las obtenidas por sistemas NER para texto en inglés general, como demuestra el hecho de que en la sexta *Message Understanding Conference* (MUC), en la que se abordan tareas de NER sobre lenguaje general, el mejor sistema obtuvo un 96 % *recall* y un 97 % *precision*, igualando el rendimiento humano [293].

Los métodos de NER pueden ser clasificados en distintos grupos:

1. Enfoques basados en diccionarios. Estas técnicas se basan en la comparación de los términos encontrados en los textos biomédicos con entradas de diccionarios construidos en base a recursos terminológicos bien definidos y ampliamente utilizados en el campo (referidos anteriormente) [311, 261].

Challenge	Best result			
	precision	recall	F-measure	Best system
BioCreative, 1A	82.8 %	83.5 %	83.2 %	[106]
JNLPBA	69.4 %	76.0 %	72.6 %	[127]
BioCreative II, GM	88.5 %	86.0 %	87.2 %	[29]

Cuadro 1.5: *Evaluaciones competitivas y resutados de los mejores sistemas de NER [254].*

2. Enfoques basados en reglas. Los enfoques basados en diccionarios habitualmente sólo pueden identificar variantes morfológicas de los términos de interés. Los enfoques basados en reglas pueden abordar un mayor rango de variaciones, incluídas aquellas que afectan a una secuencia de términos o al orden de las palabras. Este tipo de enfoque ha sido recientemente adoptado para la identificación de menciones de mutaciones en textos biomédicos [62].

3. Enfoques basados en técnicas de Aprendizaje Automático. Los enfoques basados en reglas están habitualmente limitados a patrones manuales y, por lo tanto, tienen poca capacidad de generalización ante nuevos casos o en nuevos dominios. Existe una amplia variedad de métodos de ML aplicados al NER como clasificadores naive Bayes [236], máquinas de vector soporte (SVM) [226], Conditional Random Fields (CRF) [218] y modelos de Markov (HMM) [106].

4. Enfoques híbridos. Los sistemas NER más recientes no hacen uso de una única técnica, sino que combinan distintos clasificadores basados en ML, que hacen uso de diccionarios y otros recursos terminológicos externos y efectúan un postprocesado basado en reglas para el refinamiento de los candidatos detectados, la resolución de acrónimos, etc. [274].

Los trabajos [28, 199, 187] proporcionan una revisión completa de métodos de NER aplicados a la biomedicina. La Sección 4.2 presenta una revisión de los corpora que contienen anotaciones de entidades biológicas en biomedicina.

1.3.1.2 Técnicas de text-mining para extraer relaciones entre entidades biomédicas

Existen numerosos trabajos de investigación que abordan la identificación de relaciones entre entidades biológicas a partir de la literatura biomédica. En particular, en esta sección proponemos un repaso por aquellos dedicados a la extracción de relaciones Gen-Gen, Proteína-Proteína y Gen-Enfermedad, prestando especial atención a aquellos que utilizan árboles de dependencia o analizadores sintácticos. Para una

completa revisión del estado del arte en BioNLP (Biomedical Natural Language Processing) recomendamos la consulta de algunas publicaciones de reciente aparición: [28, 351, 187, 66]. Los corpora disponibles para el entrenamiento de técnicas de ML para el reconocimiento de este tipo de relaciones se revisan en la sección 4.2.

Métodos basados en la co-ocurrencia de las entidades

Los enfoques más simples para extraer relaciones entre entidades biomédicas se basan en la co-ocurrencia de las entidades en los abstracts o los textos de las publicaciones. Estos enfoques asumen que entidades relacionadas co-ocurrirán significativamente en los textos de los artículos. Algunos trabajos en esta línea son [92], [155] y [157]. Estos métodos presentan dos inconvenientes principales. En primer lugar, dado que sólo detectan la ocurrencia de las entidades de interés (genes, proteínas o enfermedades), no proporcionan ninguna otra información de la relación extraída, como el papel de las entidades, el tipo de interacción, etc. En segundo lugar, aunque el *recall* alcanzado por estos enfoques es muy alto, el número de falsos positivos también es muy elevado, con lo que la *precision* es muy baja. Por ejemplo, los métodos de co-ocurrencia constatarían erróneamente la existencia de relaciones entre las entidades que aparecen en las siguientes sentencias:

`We genotyped patients at 110 SNPs and four repeat polymorphisms located in seven candidate genes ( HTR1A, HTR2A, HTR2C, MAOA, SLC6A4, TPH1, and TPH2 ).`

`After our research, we fail to reveal that the inhibition of C5a is due to the action of SCPA.`

Métodos basados en patrones.

Otros métodos están basados en la definición de patrones o reglas para detectar relaciones entre entidades. Estos patrones normalmente se definen en base a palabras clave en el texto (preposiciones frecuentes, *action verbs* - como *bind, inhibit, promote*, etc.), pero también se emplean *Part-of-Speech tags*, categorías sintácticas y restricciones semánticas. Estos patrones pueden estar codificados manualmente ([51], [52], [238], [77] y [272]) o aprenderse a partir de un conjunto de entrenamiento anotado ([145], [59], [258], [129], [132],[130]). El mayor inconveniente de los enfoques

con patrones codificados manualmente es que estan limitados a la detección de relaciones que hayan sido previamente definidas e incorporadas en el sistema. La generación automática de reglas o patrones también está limitada por la calidad y extensión del conjunto de entrenamiento, siendo habitual el sobre-aprendizaje de los datos de entrenamiento. El *recall* obtenido por este tipo de métodos aún es bajo.

Métodos basados en el Análisis Sintáctico de las sentencias.

El tercer grupo de enfoques se basa en el análisis sintáctico de las oraciones para desgranar los componentes y la estructura gramatical que permita la identificación de relaciones. La principal ventaja de estos métodos reside en que, al estudiar las relaciones sintácticas entre las palabras, pueden identificar estructuras con alto nivel de anidamiento y dependencias entre palabras muy distantes que serían difícules de identificar utilizando patrones superficiales. Para el análisis sintáctico pueden utilizarse métodos de *Full Parsing* o *Shallow Parsing*.

Los enfoques basados en *Shallow Parsers* ([305, 253, 240]) analizan parcialmente la estructura sintáctica de la oración e identifican dependencias entre los elementos de interés para extraer relaciones. Los métodos basados en *Full Parsers* ([263, 304, 93, 248]) inspeccionan toda la estructura sintáctica de la oración para identificar las relaciones de interés. En general, los métodos basados en *Shallow Parsing* son más rápidos y menos complejos que los de *Full Parsing*, pero éstos últimos tienen un mayor potencial, ya que manejan todas las dependencias sintácticas de las oraciones. La principal limitación de los métodos de *Full Parsing* es su sensibilidad ante errores del analizador sintáctico. Esto supone un serio inconveniente en este campo, ya que el lenguaje biomédico es complejo, con una gran variedad de vocabulario específico, y las oraciones de los artículos en este ámbito son largas y sintácticamente complejas. Esto produce que los analizadores sintácticos cometan muchos errores, e incluso que agoten sus recursos de tiempo o memoria y no puedan efectuar el análisis de muchas sentencias complejas [73].

A continuación se describen brevemente algunos de los sistemas existentes que utilizan el análisis sintáctico para extraer relaciones.

GENIES ([109]) utiliza un analizador sintáctico y una gramática semántica basada en un amplio conjunto de patrones semánticos (que también incorporan información sintáctica) para detectar las estructuras más frecuentemente utilizadas para constatar una relación. Este trabajo fue posteriormente extendido como GeneWays ([266]), que

también incluye una interfaz web que permite a los usuarios buscar y cargar artículos de interés para el análisis. Su inconveniente principal reside en que puede requerir el rediseño y la extensión de la gramática para adecuarse a dominios particulares y obtener mejores resultados. La falta de flexibilidad asociada a patrones codificados manualmente se convierte en la principal limitación de este sistema y en la causa de su bajo *recall*.

MedScan ([80]) es otro sistema que incorpora un analizador sintáctico ([26]) y una gramática libre de contexto para generar un conjunto de estructuras sintácticas que representan cada oración. A continuación, un interpretador semántico transforma estas estructuras sintácticas en una representación semántica o *marco* (del inglés, *semantic frame*). Estos marcos se definen *a priori* para las palabras clave y los verbos más comunes. Por ejemplo, uno de los marcos semánticos del verbo *inhibit* contempla la existencia de un agente y de un paciente de la acción expresada por el verbo. De nuevo, este enfoque está limitado por el uso de patrones y marcos semánticos predefinidos que habitualmente no recogen la complejidad de las frases biomédicas reales.

IntEx ([21]) utiliza el analizador sintáctico *Link Grammar* ([285]) y ontologías biomédicas para detectar entidades biológicas y sus papeles sintácticos en las frases. Este sistema extrae interacciones analizando la correspondencia entre los roles sintácticos detectados por el analizador y una serie de patrones definidos manualmente en el sistema. A continuación se muestra un ejemplo de este tipo de patrones, en este caso el patrón que modela las relaciones de tipo *Of ... by...*:
(...<InteractionWord(action)>...of...<theme>...by...<agent>...)
Este sistema ha sido extendido con un lenguaje de definición y consulta, que permite al usuario crear su propio conjunto de patrones ([122]). Sin embargo, no se han observado mejoras relevantes con la adición de patrones extra (3,5 %-7,1 % de *precision* y *recall*, respectivamente, en los resultados oficiales de BioCreAtIvE II ([122]), mientras que la adición de estos patrones implica un aumento de complejidad que afecta al rendimiento del sistema.

En un trabajo más reciente ([154]) se emplea el parser de Stanford ([182], [181]) con una gramática probabilística libre de contexto (*probabilistic context free grammar*, PCFG) para identificar relaciones proteína-proteína. Para reducir el número de errores del analizador se emplea una técnica que consiste en simplificar las frases biomédicas sustituyendo las entidades identificadas, las frases nominales

y los textos en paréntesis con códigos predefinidos. Los resultados sobre un corpus pequeño (Yapex corpus, ver sección 4.2) muestran un menor número de "abortos" del analizador gracias a estas simplificaciones, aunque no se indica la influencia en el número de errores del analizador. Con "aborto" nos referimos al hecho de que el analizador no puede analizar la sentencia y finaliza su ejecución abruptamente. Con error hacemos referencia a que la estructura sintáctica generada por el analizador no es correcta. Después de analizar las sentencias con el *parser* de Stanford, el sistema explora el árbol de dependencias generado, y busca estructuras que se correspondan con patrones predefinidos manualmente por el usuario (utilizando POS tags, etiquetas sintácticas y palabras clave).

Rel-Ex ([113]) también utiliza el parser de Stanford para generar un árbol de dependencias en el que se buscan estructuras que se correspondan con patrones predefinidos. Este sistema implementa patrones para tres tipos de relaciones: *Agente-relación-Paciente* (ej. *A activates B*), *relacion-de-Paciente-a cargo de-Agente* (ej: *Activation of A by B*) y *relación-entre-Agente-y-Paciente* (ej. *Interaction between A and B*). Estos patrones están definidos en base a etiquetas sintácticas y palabras clave.

Recientemente ha surgido una nueva familia de enfoques que utilizan funciones kernels definidas sobre árboles sintácticos y SVM para la clasificación de las oraciones y la identificación de relaciones proteína-proteína ([22, 104, 177, 335, 227, 50]). [104] utiliza el parser de Stanford y define una función kernel basada en la similitud del camino más corto en el grafo de dependencias entre la pareja de proteínas. [22] propone un kernel que recoge mayor información estructural sobre la oración, considerando todo el grafo de dependencias, aunque otorga mayor influencia (por medio de la asignación de pesos) a los tokens y dependencias del camino más corto entre las proteínas. [177] plantea una comparativa de distintos tipos de kernels definidos sobre el grafo de dependencias, alcanzando un *F-measure* de 77,5 para el corpus LLL05 [7]. [335] propone un método denominado BioPPISVMExtractor que utiliza un amplio conjunto de características de distintos niveles para representar las oraciones y realizar la clasificación para detectar relaciones proteína-proteína. Este conjunto de características incluye propiedades referentes a las palabras de la oración, palabras clave, distancia en palabras entre las proteínas y distintas características extraídas de los árboles sintácticos. [50] también propone la utilización de un amplio conjunto de características junto con kernels definidos sobre los caminos entre las dos proteínas en el grafo de dependencias. [227] propone la combinación varios kernels definidos

sobre representaciones obtenidas por distintos analizadores sintácticos para capturar toda la información posible de la oración, obteniendo resultados muy prometedores sobre varios corpora.

Los enfoques mencionados basados en *full parsing* muestran los resultados más prometedores y un rendimiento potencial muy elevado. Estos sistemas manejan toda la estructura sintáctica de las frases por lo que en teoría cuentan con toda la información necesaria para identificar relaciones entre pares de entidades biomédicas. Sin embargo, el *full parsing* de frases biomédicas completas consume mucho tiempo y los analizadores fallan con frecuencia o proporcionan dependencias incorrectas [73]. Una revisión reciente evalúa distintos analizadores sintácticos respecto a su contribución a la bondad de clasificación de un sistema de identificación de interacciones proteína-proteína [228], obteniendo los mejores resultados con la combinación de los *parsers* Enju (`http://www-tsujii.is.s.u-tokyo.ac.jp/enju`) y el *parser* propuesto por Charniak y Johnson [67].

A pesar de los crecientes esfuerzos en el campo y la aparición de un elevado número de métodos para la identificación de relaciones entre proteínas de la literatura biomédica, el rendimiento de los métodos en los distintos corpora disponibles es muy variable, su funcionamiento se ve dramáticamente afectado por el rendimiento de las herramientas de NER empleadas, y la disponibilidad de métodos sencillos de instalar y utilizar por usuarios no expertos es muy limitada [164], siendo necesario, por lo tanto, un mayor esfuerzo en estas líneas.

1.3.1.3 Sistemas software existentes

En los últimos años han surgido numerosas iniciativas académicas y privadas para la producción de herramientas y sistemas destinados a la extracción de información de la literatura biomédica y la recuperación de aquella información que resulta de interés para el usuario. En esta sección presentamos una breve descripción de estos sitemas junto con sus ventajas e inconvenientes.

PathWayStudio y MedScan de Ariadne Genomics.

PathWayStudio es un *software* diseñado para construir y analizar *pathways* en base datos biológicos y para encontrar relaciones entre genes, proteínas, procesos biológicos y enfermedades con la información extraída de la literatura. Este sistema incluye una base de datos de *pathways* supervisados por expertos, que por tanto, es muy fiable.

Sin embargo, esta base de datos sólo incluye algunas decenas de *pathways* que, además, no se han sido actualizados con información publicada recientemente. El modo de actualizar este conjunto de interacciones validadas es utilizando MedScan [80], una herramienta para extraer relaciones entre proteínas que, como ya se introdujo en la sección 1.3.1.2, sólo encuentra relaciones que se correspondan con una serie de patrones predefinidos, por lo que el *recall* obtenido es bajo.
(`http://www.ariadnegenomics.com/products/pathway-studio/`)

Ingenuity Pathways Analysis de Ingenuity Systems.

Ingenuity Systems proporciona una base de datos extensa y fiable de relaciones entre proteínas, genes, compuestos biológicos, tejidos, células, fármacos y enfermedades. Todas las interacciones recogidas en la base de datos han sido extraídas de la literatura de forma manual. A pesar de resultar un recurso de gran interés y utilidad, sólo recoge una pequeña fracción del conocimiento disponible en la literatura.
(`http://www.ingenuity.com/`)

Geneways de la Universidad de Columbia.

Este sistema se nutre de datos proporcionados por un conjunto de herramientas de Procesamiento de Lenguaje Natural Biomédico (BNLP) cuyo principal módulo es GENIE [109] (ver sección 1.3.1.2). Este sistema identifica relaciones que se corresponden con patrones predefinidos manualmente, por lo que, de nuevo, carece de flexibilidad y presenta bajo *recall*.
(`http://geneways.genomecenter.columbia.edu/`)

Chilibot.

Chilibot [68] genera consultas para PubMed con los dos genes/proteínas de interés y analiza los textos obtenidos para encontrar relaciones entre las dos entidades. Este análisis se basa en reglas que clasifican las frases en categorias: estimulación, inhibición, neutra, paralela o sólo co-ocurrencia. Estas reglas se fundamentan en la presencia o ausencia de ciertos verbos clave (*activate, facilitate, increase, induce, stimulate*, etc.) entre las dos entidades. Aunque el sistema es muy sencillo de utilizar, intuitivo y siempre actualizado (dado que PubMed es directamente consultado en cada ejecución de Chilibot), el módulo de Procesamiento de Lenguaje Biomédico resulta demasiado simple para capturar ciertos tipos de relaciones más complejas, y presenta

numerosos falsos positivos.
(http://www.chilibot.net)

iHOP.

iHOP [142] es un sistema que permite navegar a través de textos de PubMed, en los que un gen o proteína de interés co-ocurre con otra entidad biomédica. El sistema genera consultas para PubMed y analiza los resultados buscando co-ocurrencia del término de interés con otro gen o proteína. El motor de Procesamiento del Lenguaje es simple y se basa principalmente en la búsqueda de palabras clave (*action verbs*) y la correspondencia con patrones sencillos predefinidos. La principal ventaja de este software es que permite al usuario navegar por los extractos de texto obtenidos y acceder a la información de los genes y proteínas que se asocian a la entidad de interés (las entidades biomédicas aparecen como hipervínculos entre textos).
(http://www.ihop-net.org/UniPub/iHOP/)

PubNet.

PubNet [99] es otro sistema que permite extraer distintos tipos de relaciones de los resultados de consultas en PubMed y muestra estas relaciones gráficamente en forma de redes, permitiendo al usuario navegar cómodamente por los resultados y efectuar análisis más detallados de los textos que soportan las relaciones e incluso de la topología de la red obtenida.
(http://pubnet.gersteinlab.org/)

AliBaba.

AliBaba [249] efectúa consultas en PubMed para identificar relaciones entre distintos tipos de entidades biológicas: genes y proteínas, especies, tejidos, localizaciones celulares, fármacos y enzimas. La identificación de relaciones se basa en simple co-ocurrencia de las entidades detectadas, aunque para extraer relaciones proteína-proteína se utilizan patrones predefinidos que identifican los roles (agente, paciente) de las entidades involucradas. Las relaciones extraídas se muestran gráficamente al usuario en un grafo de interacciones.
(http://alibaba.informatik.hu-berlin.de/)

BotXMiner.

BotXMiner [231] extrae información de textos de artículos recuperados por MedLine. Este sistema busca en el título, abstract y términos MeSH las palabras clave que representan las entidades biomédicas. Los artículos recuperados son agrupados y mostrados al usuario gráficamente en un grafo en el que las entidades de interés (ej. genes, proteínas) son nodos y las citaciones son arcos. Un arco entre dos nodos se colorea en función del número de co-ocurrencias en diferentes artículos de las dos entidades conectadas por el citado arco.
(`http://botdb.abcc.ncifcrf.gov/botXminer/`)

STRING.

STRING [322] es una base de datos de interacciones proteína-proteína extraídas de distintas fuentes de datos, principalmente de PubMed. STRING recibe como entrada una lista de proteínas, y genera un grafo en el que los nodos se corresponden con las proteínas, y los arcos con las interacciones entre las mismas, sustentadas por distintas fuentes biológicas (experimentales, literatura, etc). Para las evidencias textuales, STRING proporciona la lista de artículos que sustentan la relación, junto con los extractos de texto en los que esas proteinas co-ocurren. El motor de BLP de esta herramienta es muy sencillo y se basa principalmente en la co-ocurrencia significativa de las proteínas.
(`http://string.embl.de`)

AKANE++

AKANE++ es un sistema de identificación de relaciones proteína-proteína a nivel de frases. Este sistema procesa los abstracts en un *pipeline* que incluye la detección de tokens y oraciones y el análisis sintáctico de las mismas. Un sistema de NER identifica proteínas y las normaliza contra entradas de Uniprot. Las parejas de proteínas que co-ocurren en una oración, son consideradas candidatas a relaciones proteína-proteína. Estas parejas candidatas son pasadas a un clasificador que utiliza kernels sobre la estructura sintáctica de las oraciones para determinar si existe realmente una relación entre ellas [227]. (`http://www-tsujii.is.s.u-tokyo.ac.jp/~satre/akane/`)

PIE

PIE (Protein Interaction Extraction) [176] es un sistema web que permite identificar interacciones proteína-proteína en la literatura biomédica. El sistema está conforma-

do por dos módulos. El primero efectúa un filtrado de artículos candidatos a contener relaciones proteína-proteína, implementando un clasificador Naive Bayes que utiliza el conjunto de palabras de un artículo (*bag-of-words*) como representación del mismo para su clasificación. El segundo módulo lleva a cabo un filtrado de frases candidatas a contener estas relaciones, implementando un kernel de convolución [74] para calcular la similitud de las estructuras gramaticales de las frases candidatas con las de los ejemplos positivos del conjunto de entrenamiento. La clasificación de las frases candidatas se lleva a cabo por medio de una SVM. (`http://pie.snu.ac.kr/`)

EBIMed y Protein Corral

EBIMed y Protein Corral son dos plataformas del *European Bioinformatics Institute* (EBI) para la recuperación de artículos de Medline y la extracción de información de los mismos. Protein Corral se centra en el análisis del texto para identificar relaciones entre genes y proteínas normalizadas contra Uniprot. Las relaciones son identificadas por distintas técnicas: co-ocurrencia de las proteínas, co-ocurrencia de las proteínas y de un *action verb* en la oración, y correspondencia con patrones definidos manualmente. El máximo nivel de confianza es el otorgado por los patrones manuales, con lo que esta herramienta presentará los mismos defectos que otros métodos basados en patrones anteriormente comentados. EBIMed extiende Protein Corral y recoge un abanico más amplio de entidades: proteínas, genes, términos de GO, nombres de fármacos y especies, identificando relaciones entre ellos utilizando las mismas técnicas que Protein Corral.

(`http://www.ebi.ac.uk/Rebholz-srv/pcorral` y
`http://www.ebi.ac.uk/Rebholz-srv/ebimed/`)

U-compare

U-compare [167] es una plataforma web que permite al usuario diseñar un *pipeline* de procesamiento de textos a medida, utilizando distintos módulos de text-mining y corpora disponibles. Está construido sobre la plataforma UIMA (*Unstructured Information Management Architecture*). La principal aportación de U-compare es que permite ejecutar y comparar distintas herramientas de text-mining, obteniendo estadísticas de rendimiento y la posibilidad de visualizar los resultados. (`http://u-compare.org/`)

BioCreative Meta-Server.

El Meta-servidor de BioCreative [198] es una plataforma prototipo que integra los servidores de distintos grupos de investigación participantes en las evaluaciones competitivas de Biocreative. Este meta-servidor proporciona simultáneamente anotaciones para abstracts de PubMed/Medline generadas por medio de estos sistemas automáticos. Las entidades anotadas incluyen nombres de genes, identificadores de genes, especies y relaciones proteína-proteína. Este meta-servidor incluye algunos de los recursos anteriormente descritos, como AKANE++, PIE o AliBaba, entre otros. Presenta la ventaja de que permite utilizar, simultáneamente, distintos métodos de reconocimiento de entidades y relaciones y visualizar los resultados. Además, presenta una API para efectuar consultas y recuperar los resultados de forma automática. (`http://bcms.bioinfo.cnio.es/`)

Bases de datos con interacciones proteína-proteína

En los últimos años han surgido numerosas bases de datos que contienen interacciones entre proteínas: BIND (`http://www.bind.ca`), DIP (`http://dip.doe-mbi.ucla.edu`),
HPRD (`http://www.hprd.org`), MIPS (`http://mips.gsf.de/proj/ppi/`),
REACTOME (`http://www.reactome.org/`), MINT (`http://mint.bio.uniroma2.it/mint`) o IntAct (`http://www.ebi.ac.uk/intact/main.xhtml`), entre otras. Todas estas bases de datos se poblan con conocimiento extraído manualmente de la literatura especializada, por lo que sólo cubren una pequeña fracción de todas las interacciones entre proteínas conocidas. El trabajo de [214] señala, además, que el solapamiento entre algunas de estas bases de datos es muy bajo, demostrando que los procesos manuales de extracción de conocimiento no son suficientes para procesar la ingente cantidad de conocimiento que se publica esparcida por cientos de revistas diferentes cada día. Por poner un ejemplo, un estudio del equipo de mantenimiento de la base de datos BIND, estima que se publican alrededor de 1900 interacciones cada mes, esparcidas en unas 80 revistas de índice de impacto [24].

1.3.2 Análisis automático de historias clínicas.

Además de la literatura biomédica, los historiales clínicos constituyen una rica y extensa fuente de información textual para la investigación médica [252]. Este conocimiento se almacena principalmente en forma de texto libre, y contiene abundante información médica relacionada con síntomas, diagnósticos, tratamientos e historias clínicas. El valioso contenido de estos registros ha provocado un creciente interés en los últimos años en la aplicación de técnicas de Aprendizaje Automático para la extracción de conocimiento de estas fuentes.

Sin embargo, el texto de los historiales clínicos es inherentemente desestructurado, redundante, ambiguo e incluso gramaticalmente incorrecto [247]. Además contiene gran cantidad de acronismos, abreviaciones y utiliza un lenguaje altamente especializado. La mayoría de los historiales clínicos incluyen información relativa al diagnóstico principal, diagnósticos secundarios, seguimiento de la enfermedad, seguimiento de la medicación prescrita para el paciente, historia médica anterior, etc. En particular, a continuación se enumeran algunos de los epígrafes más comunes en los textos de historiales clínicos escritos en lengua inglesa:

- Principal Diagnoses
- Secondary Diagnoses
- History of present illness
- Pre-Admission medications
- Past medical history
- Family history
- Social history
- Allergies
- Admission physical examination
- Studies
- Procedure
- Laboratory data
- Hospital course by problem
- Complications
- Consultants

Principal Diagnosis: Bacteremia , endocarditis. Secondary Diagnosis: Status post aortic valve repair. Hypertension. Coronary artery disease status post CABG. Anemia. Status post hemorrhagic stroke. History of present illness: This is an 82-year-old male with history of multiple medical problems including recent aortic valve replacement for aortic stenosis on 4/2 , CAD status post CABG , CHF , atrial fibrillation with slow ventricular response , insulin-dependent diabetes who presents from rehab with positive blood cultures. Medications on admission: 1) Vasotec , 10 mg b.i.d.; 2) Mevacor , 20 mg q d; 3) Imuran , 150 mg q day. Alergies: NKA Family history: The patient's mother had a myocardial infarction in her 60's. Social history: He smoked one pack per day. Denies alcohol use. Physical examination: Afebrile , pulse irregular , 125; blood pressure 140/100. Pulse oximetry was 97 % on 2 liters. General: This is an elderly male in no acute distress. Neck: Supple , no lymphadenopathy , no thyroid enlargement , JVP at 18 cm with flutter waves. Carotids are 2+ bilaterally. Pulmonary: Bibasilar rales. Cardiovascular: Irregular rate and rhythm , tachycardic , S1 , S2 normal. No murmurs , rubs or gallops.

Cuadro 1.6: *Extractos de historiales clínicos reales.*

- Physical examination on discharge
- Discharge medication
- Disposition
- Follow-up appointments
- Code status

La Tabla 1.6 muestra extractos de historias clínicas reales, que permiten ilustrar algunas de sus características anteriormente citadas: terminología muy específica (nombres comerciales de medicamentos, nombres de marcadores diagnósticos, términos anatómicos y médicos, etc.), uso extendido de acronismos (mg b.i.d., CABG, CAD, CHF, JVD), oraciones gramaticalmente incompletas o incorrectas (por ejemplo, oraciones sin verbo). Estas características dificultan el análisis automático de este tipo de textos mediante técnicas de Aprendizaje Automático y Procesamiento de Lenguaje Natural. Además, la normativa sobre privacidad y protección de datos de carácter personal hace que la publicación de este tipo de corpora para la investigación en este campo sea muy escasa [317]. Es por ello que los esfuerzos recientes para la difusión para investigación de corpora sobre historiales clínicos reales resultan muy valiosos ([247, 319, 290, 316]).

Numerosos métodos de Aprendizaje Automático y Procesamiento de Lenguaje Natural se han aplicado sobre los historiales clínicos para realizar tareas de reconocimiento de entidades (NER) y ocultación de datos personales [317, 328, 234, 318], y clasificación de textos [239, 197]. El capítulo 5 se centra, en particular, en la clasificación de textos. Concretamente, el interés de nuestra investigación consiste en recopilar suficiente información del texto de un historial clínico para asignarle, de forma automática, el diagnóstico correcto.

1.4 Sistemas de Anotación de textos biomédicos

Las técnicas de Procesamiento de Lenguaje Natural (NLP) han sido aplicadas con éxito en campos que necesitan extraer información de textos escritos en lenguaje natural [28]. Los *corpora* (en singular, *corpus*) son colecciones de textos de un tipo específico o relativos a un ámbito concreto, que proporcionan el material de referencia a partir del cual entrenar y evaluar los métodos de NLP. Estas colecciones *son anotadas* para hacer explícitas las entidades o estructuras que existen en el texto y resultan de interés para una determinada tarea o problema. Estas entidades y estructuras textuales de interés definen lo que se denomina como *esquema de anotación*.

Las anotaciones pueden ser efectuadas por expertos humanos (*curators*) o de forma automática, utilizando herramientas de text-mining y NLP. Las anotaciones humanas resultan de enorme valor, puesto que son precisas y de alta calidad, y los anotadores están entrenados para reformatear los hechos biológicos de interés para que se adecúen al esquema de anotación [261].

Esta sección revisa los sistemas de anotación de textos biomédicos existentes y destaca el papel emergente de los sistemas de anotación colaborativa y distribuida en las ciencias de la vida.

1.4.1 Anotación social y colaborativa en las ciencias de la vida

En los últimos años han sugido diversas herramientas de anotación de textos de propósito general, como Knowtator [11], WordFreak [14], SAFE-GATE [79] o iAnnotate [10]. Estas herramientas proporcionan al usuario mecanismos flexibles para definir el esquema de anotación, por lo que pueden ser adaptadas a tareas específicas de anotación de textos biomédicos. Algunos grupos de BioNLP han creado *ad-hoc* sus propias herramientas para la anotación de corpora específicos con esquemas de anotación prefijados. Por ejemplo, Xconc Suite es una herramienta para anotar eventos en el corpus GENIA [174].

Estas herramientas están fundamentalmente diseñadas para esfuerzos de anotación que involucran a un pequeño número de anotadores bien entrenados, que realizan tareas de anotación muy específicas y conforme a esquemas de anotación complejos.

Sin embargo, en los últimos tiempos, con el auge de Internet y las comunidades virtuales, los proyectos colaborativos a gran escala están recibiendo una mayor difusión, aceptación y apoyo. Este tipo de esfuerzos muestran un enorme potencial, ya

que permiten a millones de usuarios de todo el mundo colaborar en el desarrollo de un proyecto determinado, en este caso, la anotación de textos biomédicos. Las herramientas referidas anteriormente no están diseñadas para soportar este tipo de esfuerzos distribuidos de anotación a gran escala.

Sistemas web como Delicious (`www.delicious.com`), o Connotea (`www.connotea.org`) facilitan el etiquetado de recursos online y referencias bibliográficas. Este tipo de recursos sacan partido del conocimiento comunitario disponible gracias al etiquetado colectivo. Además, facilitan a los usuarios encontrar e interaccionar con otros usuarios con sus mismos intereses y problemas (dos usuarios serán similares si anotan los mismos tipos de objetos con etiquetas similares), por ejemplo, usuarios cuya investigación se centra en el mismo gen o metodología.

La plataforma Mechanical Turk (AMT) de Amazon Web Services (AWS, `http://aws.amazon.com/`) es otro ejemplo de servicio basado en esfuerzos distribuidos y comunitarios. Se basa en la premisa de que una comunidad de anotadores, no experta en una materia, puede proporcionar anotaciones de la misma calidad que un pequeño grupo de anotadores expertos, con las ventajas que ello conlleva en reducción de costes, ya los anotadores expertos son más caros. Para que esta premisa sea cierta en tareas referentes a NLP, es necesario que la tarea propuesta pueda ser llevada a cabo por cualquier parlante nativo en la lengua en la que la tarea se plantea, es decir, no debe requerir más entrenamiento que la comprensión del lenguaje. En un trabajo reciente, [289] demuestra la efectividad del uso de AMT para varias tareas de anotación relacionadas con problemas clásicos de NLP: reconocimiento de emociones, similitud de palabras, inferencia de hipótesis de un texto, ordenación cronológica de hechos relatados en un texto y resolución de ambigüedades. En este trabajo se constata que, con un pequeño número de anotaciones efectuadas por no-expertos, se iguala el rendimiento obtenido por un clasificador entrenado con anotaciones efectuadas por expertos. Por ejemplo, para el problema de reconocimiento de emociones, bastan cuatro anotaciones de no-expertos (en promedio) por cada item para emular la calidad de anotación de un experto, con unos costes mucho menores. Tras este artículo pionero en la materia, han surgido otros trabajos muy recientes que proponen y evaluan el uso de AMT para la anotación colaborativa de corpora y el entrenamiento de sistemas de ML [259, 97, 60, 63].

La anotación colaborativa o comunitaria también ha sido adoptada recientemente por la comunidad biomédica. Por ejemplo, WikiProteins [229] o WikiGene [211] proporcionan entornos apropiados para la anotación comunitaria de genes y proteinas, entre otras entidades biológicas de interés, lo que permite a la comunidad científica

beneficiarse directamente, y con mínimo coste, de conocimiento generado y revisado por la propia comunidad. La sección 4.3 describe con más detalle éstas y otras herramientas de anotación colaborativa biomédica.

La industria editorial biomédica también ha realizado recientemente esfuerzos para adoptar y promover redes sociales. BioMedExperts (BME, `http://www.biomedexperts.com`) es una red social dirigida a investigadores, en la que las referencias bibliográficas y la co-autoría de trabajos de investigación se utilizan para promover y sustentar interacciones entre los usuarios. Aunque este sistema no permite añadir etiquetas a los usuarios, realiza un etiquetado automático basado en una terminología de referencia, permitiendo identificar investigadores con intereses similares. La red Nature Network (`http://network.nature.com/`) funciona de forma similar, aunque tanpoco permite utilizar ontologías o terminologías para anotar las referencias bibliográficas.

1.4.2 La Web Semántica en las ciencias de la vida.

El avance de la investigación en biomedicina depende en gran medida de la disponibilidad de información y la utilización de la misma de forma eficaz. Las ciencias biomédicas o ciencias de la vida se han convertido campos en los que se dispone de cantidades masivas de información, y existe, por tanto, la necesidad de combinar datos de distintas fuentes para analizarlos de manera más efectiva y extraer conocimiento de los mismos. La Web Semántica ofrece un marco social y técnico para la integración y distribución de conocimiento biomédico a través de la Web [345]. Se trata de una iniciativa que propone describir la semántica de los contenidos de la Web actual mediante metadatos, de forma que sea posible buscar, recuperar e integrar información automáticamente. Para ello se requiere de la utilización de lenguajes de representación, protocolos web y tecnologías estándares para la integración de información.

Grandes centros de investigación dedicados a la Bioinformática como el Instituto Europeo de Bioinformática (http://www.ebi.ac.uk) o el *Nacional Center for Biotechnology Information* de EEUU (http://www.ncbi.nlm.nih.gov) proporcionan acceso a más de 200 recursos biológicos. En este contexto, los enlaces entre distintos recursos y bases de datos constituyen la base fundamental para la integración de datos, pero la ausencia de una representación común para los datos y de enlaces entre los mismos hace que la integración de los distintos recursos sea altamente costosa. Recientemente importantes bases de datos como Uniprot [36] han comenzado a distri-

buir sus datos en formato RDF. El formato RDF (Resource Description Framework, http://www.w3.org/RDF/) es un formato estándar propuesto por el *World Wide Web Consortium's Semantic Web activities* (http://www.w3.org/2001/sw/) para entornos de trabajo distribuidos y descentralizados. El modelo de datos RDF representa cualquier tipo de información en forma de declaraciones sujeto-predicado-objeto. Para permitir el enlazado de datos en la web, RDF impone que cada elemento tenga un identificador global único, que permita a cualquier sistema formular declaraciones sobre el mismo. La utilización de estos identificadores globales y la estructura del modelo de datos RDF permiten la integración de declaraciones efectuadas por distintos recursos y bases de datos y, de este modo, la posibilidad de efectuar consultas entre distintos recursos. RDF constituye, por lo tanto, un elemento fundamental para la anotación distribuida y la integración de datos en bioinformática.

El proyecto Bio2RDF (http://bio2rdf.org) es un ejemplo de aplicación de tecnologías de la Web Semántica sobre recursos biomédicos disponibles a través de la web. Este proyecto propone la construcción de un espacio de conocimiento común constituido por documentos RDF interrelacionados entre sí mediante URIs normalizadas y compartiendo ontologías comunes [45].

A medida que la Web Semántica se introduce en las ciencias de la vida, se extiende y profundiza el desarrollo de nuevos recursos íntimamente asociados a estas tecnologías, tales como las bio-ontologías. Las bio-ontologías proporcionan conocimiento esencial en el domino del problema y resultan fundamentales para una integración efectiva de datos, para la recuperación de información, la anotación de textos, el procesamiento de lenguaje natural y la toma de decisiones, entre otros. De este modo, numerosas ontologias están siendo desarrolladas y perfeccionadas para formalizar el conocimiento en el ámbito biomédico [280].

El *Linking Open Drug Data Task*(LODD) [159], enmarcado en el *W3C's Semantic Web for Health Care and Life Sciences Interest Group*, ha recopilado recursos relacionados con fármacos y realizado un estudio sobre cómo interrelacionar la información proporcionada por estos recursos. El proyecto ha merecido recientemente el primer premio del *Linking Open Data Triplification Challenge*, lo que muestra la importancia que la normalización de la información y la interconexión de recursos mediante aplicaciones de Web Semántica está adquiriendo en las ciencias de la vida.

La agregación de datos heterogéneos, la anotación y distribución de descubrimientos, la expresión de modelos ricos y bien-definidos para la agregación y búsqueda de información, la reutilización más sencilla de información y la aplicación de la lógica para la inferencia de nuevos descubrimientos, son sólo algunos de los beneficios que

la implantación de la Web Semántica promete. En el capítulo 4.6.3 presentamos una herramienta desarrollada con el objetivo de trasladar algunos de estos beneficios al campo de la distribución y consulta de anotaciones sobre textos biomédicos.

Parte II

Análisis de microarrays mediante Clustering y Biclustering

Capítulo

2

Clustering y biclustering para identificar patrones de máxima varianza en matrices de expresión genética

2.1 Motivación y objetivos

El Clustering es una de las técnicas más extendidas para el análisis de las matrices de expresión genética obtenidas a partir de los experimentos de microarrays [160, 169](ver Sección 2.2). Agrupando conjuntamente los genes que presentan el mismo comportamiento en distintas circunstancias y condiciones experimentales, los clusters resultantes pueden ayudar en la identificación de módulos funcionales. Los enfoques clásicos de clustering agrupan los genes en clusters exclusivos, no permitiendo solapamiento entre clusters. En un sistema biológico real, un gen puede desempeñar diversas funciones y participar en distintos procesos biológicos, por lo que resulta deseable que un mismo gen pudiera pertenecer a distintos agrupamientos. Para solucionar este problema han surgido numerosas propuestas entre las que cabe destacar, por su repercusión en trabajos posteriores, algoritmos como Gene Shaving [135] y los basados en técnicas difusas [47, 86, 44].

Gene Shaving es un método de clustering que identifica clusters coherentes (con genes muy similares entre sí) y con alta varianza a lo largo de las condiciones, permitiendo solapamiento entre clusters. Se trata de un algoritmo desarrollado en la Universidad de Stanford que ha sido ampliamente utilizado para el análisis de datos

de expresión genética, con más de 350 citas del artículo original [1] y modificaciones propuestas recientemente [347]. La aportación principal de Gene Shaving reside en que no sólo tiene en cuenta la similitud entre los patrones de expresión de los genes (o coherencia del cluster), sino que, además, considera la varianza de la expresión de los genes para las muestras de la matriz de expresión. De este modo, el objetivo es encontrar patrones que involucren genes que actúen conjuntamente y presenten un comportamiento muy diferente a lo largo de las muestras, ignorando los genes que no participen en ningún proceso biológico activo, así como los que estén activados a nivel constante para todas las muestras. Los clusters obtenidos resultan, por tanto, muy útiles para caracterizar los distintos tipos de muestras, así como los procesos biológicos subyacentes que pueden producir esta diferencia de comportamiento para los grupos de genes.

Sin embargo, los algoritmos de clustering presentan una limitación al ser aplicados al análisis de matrices de expresión genética, ya que estas matrices típicamente contienen un elevado número de muestras (del orden de decenas o cientos), que además, pueden ser muy heterogéneas (por ejemplo, muestras de pacientes con distintas patologías). En este contexto, exigir que un conjunto de genes presente un comportamiento similar para todas las muestras en estudio puede ser un criterio demasiado restrictivo o poco realista, que no nos permite detectar otros patrones interesantes.

El biclustering ha surgido como solución ante este problema [133], ya que permite identificar conjuntos de genes (muestras) que exhiben un comportamiento similar para un subconjunto de las muestras (genes), pero no, necesariamente, para todas ellas. De este modo, el biclustering se ha convertido en un complemento al clustering muy extendido para el análisis de matrices de expresión.

La línea de investigación que se describe en este capítulo tiene como objetivo desarrollar nuevos algoritmos no exclusivos de clustering y biclustering para la identificación de patrones potencialmente solapados que se corresponden con la definición de cluster de Gene Shaving: grupos de genes con el mismo comportamiento y alta varianza entre las muestras. En esta línea, describimos cuatro nuevos métodos no-exclusivos de clustering y biclustering: EDA-Clustering y EDA-Biclustering, basados en Algoritmos de Estimación de Distribuciones de probabilidad (EDA), GA-Clustering, basado en Algoritmos Genéticos y Gene-&-Sample Shaving, un algoritmo de bicluster basado en Análisis de Componentes Principales.

[1] Datos de Google Scholar en Diciembre de 2009.

En este capítulo se presentan los resultados obtenidos por los algoritmos desarrollados, junto con Gene Shaving, sobre dos *datasets* reales: un *dataset* del ciclo celular de *S. cerevisiae* y otro de muestras de linfomas en *H. Sapiens*. La evaluación comparativa de los resultados de los distintos algoritmos se ha realizado teniendo en cuenta la calidad y tamaño de los patrones identificados. Además, se ha empleado la ontología *Gene Ontology* [32] para evaluar la interpretación biológica de los resultados obtenidos por cada método, calculando la significación estadística de los procesos biológicos asociados a los genes de los distintos agrupamientos.

El capítulo se estructura como sigue. Las dos primeras secciones (2.2, 2.3) repasan los trabajos previos en clustering y biclustering aplicados al análisis de la expresión genética de datos de microarrays. La sección 2.4 describe el algoritmo Gene Shaving y presenta una descripción razonada de sus limitaciones (2.5). La sección 2.6 presenta dos nuevos algoritmos de clustering basados en algoritmos evolutivos: GA-Clustering y EDA-Clustering. En 2.7 se presentan dos nuevos algoritmos que extienden el marco propuesto por Gene Shaving al biclustering: EDA-biclustering, basado en algoritmos EDA, y *Gene-&-Sample Shaving*, basado en el Análisis de Componentes Principales. La sección 2.8 presenta los resultados obtenidos por los distintos algoritmos de clustering y biclustering propuestos, junto con el algoritmo Gene Shaving, en dos casos de estudio reales. Finalmente, se presentan las conclusiones obtenidas de esta línea de trabajo (2.9).

2.2 Aplicación del clustering al análisis de microarrays

Los algoritmos de clustering agrupan conjuntamente genes (condiciones) con niveles de expresión similares para las condiciones (genes). De este modo, los genes pertenecientes a un mismo cluster responden de manera similar a diferentes condiciones y circunstancias, por lo que probablemente desempeñan la misma función biológica, o funciones biológicas relacionadas [321].

El potencial del clustering para identificar patrones en datos de microarrays fue mostrado por primera vez por Eisen *et al.* en [103], que utiliza un algoritmo de clustering jerárquico para identificar grupos funcionales de genes (ver Figura 2.1). Aunque los dendrogramas obtenidos de los algoritmos de clustering jerárquico permiten presentar los resultados de forma intuitiva al usuario, existen 2^{n-1} ordenaciones lineales distintas para n genes, que son consistentes con la estructura del árbol. Esto debe ser

cuidadosamente considerado cuando se proceda a la elección del número de clusters, ya que cortar el dendrograma a un nivel puede llevar a perder información importante representada en otros niveles [169]. Tororen [307] presenta un método para analizar dendrogramas y determinar la altura óptima de corte utilizando clases de genes tomadas de bases de datos especializadas. Este método busca correlaciones óptimas entre las clases de genes y los clusters obtenidos con distintas alturas de corte. Bar *et al.* [40] presenta un algoritmo de clustering aglomerativo para el análisis de datos de expresión que produce un árbol k-ario (cada nodo interior del árbol presenta, al menos, k hijos) en lugar de los dendrogramas binarios. Esto permite unir directamente hasta k genes o grupos en cada nivel del árbol, para lo cual se requiere un parámetro adicional del usuario que determine el grado de similitud mínimo requerido para realizar la fusión.

Sin embargo, estos métodos no pueden corregir la naturaleza *greedy* de los algoritmos jerárquicos aglomerativos de clustering, donde un error de agrupamiento al inicio del proceso no puede ser corregido y puede afectar seriamente al resto del proceso [169]. Los trabajos [337, 120] muestran que, a pesar de su uso extendido, el rendimiento de los algoritmos de clustering jerárquico es mucho menor que el de otras técnicas clásicas de clustering como k-medias o los mapas auto-organizativos (*self-organizing maps*, SOM). Recientemente, sin embargo, Bhattacharya *et al.* [48] propone un algorimo jerárquico de clustering que estima el valor óptimo para el número de clusters y que puede recuperarse de errores de agrupamiento producidos en fases tempranas del algoritmo, mejorando los resultados obtenidos por k-medias y otros algoritmos particionales de clustering. Otras variantes de algoritmos jerárquicos han sido propuestas recientemente [237].

Además de algoritmos de clustering jerárquico, muchos otros métodos han sido empleados para el clustering de datos de expresión, por ejemplo el algoritmo k-medias [82] (y algunas variantes [34, 309]), mapas de kohonen [295, 119], árboles auto-organizativos [139, 206] y métodos probabilísticos y de teoría de grafos[281]. Del mismo modo, algunos algoritmos de clustering han sido optimizados para reducir su consumo de tiempo para el análisis de matrices de expresión, habiendo surgido nuevas variantes más eficientes [189], o que utilizan entornos especiales de computación (como computación paralela [101] o aceleración utilizando *hardware* gráfico [344]).

Los Algoritmo Evolutivos (*Evolutionary algorithms*, EA) también han sido aplicados al clustering de matrices de expresión genética. Los algoritmos evolutivos se basan en los principios de la evolución natural (selección natural, recombinación genética y mutación). Estos algoritmos parten de una población de soluciones posibles

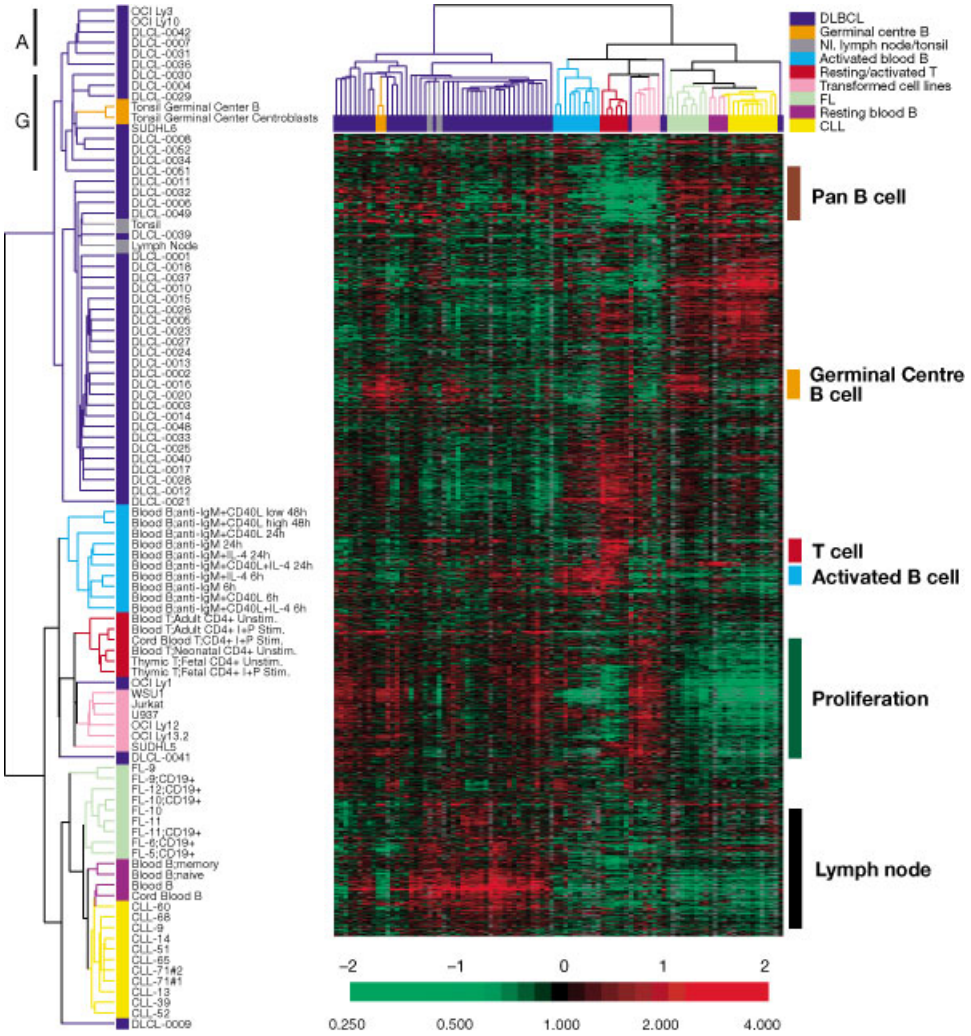


Figura 2.1: *Ejemplo de la aplicación de clustering jerárquico para agrupar muestras en una matriz de expresión. Figura tomada de [25].*

(individuos o cromosomas) para evolucionarla, generación tras generación, hacia óptimos cercanos al óptimo global. Una función de calidad o *fitness* proporciona a cada individuo cierta probabilidad de supervivencia hasta la siguiente generación. GeneClust [90] implementa un tipo de algoritmo evolutivo denominado Algoritmo Genético (*Genetic Algorithm*, GA) para identificar clusters en matrices de expresión. GeneClust utiliza como función de *fitness* la varianza total del cluster (*total within cluster variance*), que se desea minimizar, en este caso, para obtener clusters de genes con comportamiento similar. Existen algunas aproximaciones que extienden este modelo, por ejemplo, [205] propone un enfoque híbrido que combina el k-medias y los algoritmos genéticos y [39] propone un algoritmo genético multiobjetivo (MOGA) para

identificar clusters en matrices de expresión.

Revisiones completas de algoritmos de clustering aplicados al análisis de microarrays pueden consultarse en [160, 169].

Todos los enfoques mencionados agrupan los genes en clusters exclusivos, no permitiendo solapamiento entre clusters. En un sistema biológico real, un gen puede desempeñar diversas funciones y participar en distintos procesos biológicos, por lo que resulta deseable que un mismo gen pudiera pertenecer a distintos agrupamientos. Para solucionar este problema han surgido numerosas propuestas entre las que cabe destacar, por su repercusión en trabajos posteriores, los algoritmos Gene Shaving [135] (descrito en 2.4) y los basados en técnicas difusas (revisados en 3.2).

2.3 Aplicación del biclustering al análisis de microarrays

El concepto de bicluster (también denominado co-cluster o cluster *two way*) fue aplicado al análisis de datos de microarrays por primera vez en [69]. Desde entonces, numerosos algoritmos de biclustering han sido propuestos para el análisis de matrices de expresión genética [210]. Los algoritmos de biclustering permiten identificar conjuntos de genes (muestras) que exhiben un comportamiento similar para un subconjunto de las muestras (genes), pero no, necesariamente, para todas ellas. Esto resulta de un gran interés en el análisis de matrices de expresión genéticas, ya que un proceso celular de interés puede no estar activo en la totalidad de las muestras [194, 88]. Las matrices de expresión típicamente contienen un elevado número de muestras (del orden de decenas o cientos), y además, estas pueden ser muy heterogéneas (por ejemplo, muestras de pacientes con distintas patologías). En este contexto, exigir que un conjunto de genes presente un comportamiento similar para todas las muestras en estudio puede ser un criterio demasiado restrictivo o poco realista, que no nos permite detectar otros patrones interesantes. Un ejemplo que puede ilustrar esta situación se presenta en la Figura 2.2. De este modo, el biclustering se ha convertido en una alternativa al clustering muy extendida para el análisis de matrices de expresión.

Sea $A_{n\times m}$ una matriz de expresión de $n \times m$ valores reales, donde las filas representan genes, las columnas muestras, y a_{ij} representa el nivel de expresión del gen i en la muestra j. En [69], se propone el residuo cuadrático medio (*mean squared*

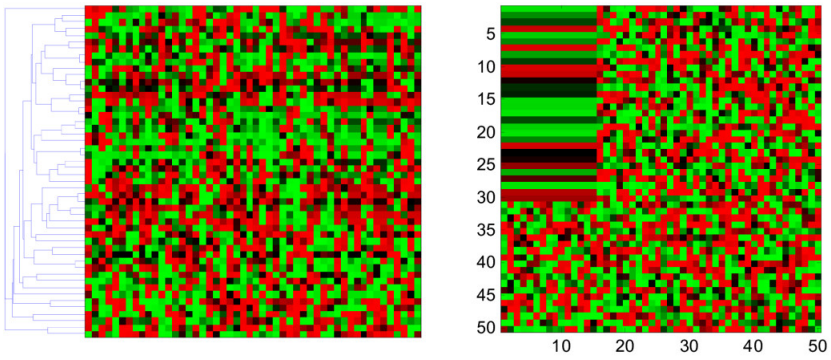


Figura 2.2: *Ejemplo ilustrativo de la utilidad del biclustering para encontrar patrones que el clustering no detecta. (a) Matriz de expresión en la que no se aprecian patrones de interés después de la aplicación de un algoritmo de clustering jerárquico. (b) Matriz de expresión en la que se han reordenado algunas filas y columnas para mostrar un patrón oculto en los datos (bicluster). Figura tomada de [114].*

residue, MSR) para medir la coherencia de los genes y muestras en un bicluster. De acuerdo con el modelo aditivo (ver Sección 1.2.3):

$$a_{ij} = \lambda + \phi_i + \gamma_j$$

si hacemos $\lambda = a_{IJ}$ (la media de la submatriz (I,J)):

$$\phi_i = a_{iJ} - a_{IJ}$$

$$\gamma_j = a_{Ij} - a_{IJ}$$

se obtiene que un bicluster coherente perfecto satisface:

$$a_{ij} = a_{iJ} + a_{Ij} - a_{IJ}$$

En biclusters no perfectos, la expresión a la izquierda de la igualdad será diferente a la de la derecha en una cantidad denominada *residuo*:

$$r(a_{ij}) = a_{ij} - a_{iJ} - a_{Ij} + a_{IJ}$$

El residuo total de un bicluster (MSR), definido por las filas I y las columnas J es la media de los residuos al cuadrado de todos los elementos del bicluster:

$$MSR(I,J) = \frac{1}{|I|\,|J|} \sum_{i \in I, j \in J} r(a_{ij})^2 \tag{2.1}$$

Una submatriz (I,J) es un δ-bicluster si $H(I,J) \leq \delta$ para algún $\delta > 0$.

Numerosos métodos tienen como objetivo la identificación de δ-biclusters mediante aproximaciones heurísticas, por ejemplo, el método de Cheng y Church [69], o los algoritmos propuestos por Yang and Wang: δ-cluster [324], p-δ-cluster [323] y FLOC [334]. Sin embargo, estas técnicas presentan ciertas limitaciones. Primero, al ser algoritmos greedy iterativos (todos excepto [323]), habitualmente convergen en óptimos locales y pueden perder biclusters importantes [346]. Segundo, el establecimiento de un valor adecuado para el umbral δ no es trivial y requiere de cierto conocimiento a priori que depende del *dataset* [303].

Cho et al. [70] introduce otro algoritmo greedy que busca una partición de la matriz de expresión en k clusters de filas y l clusters de columnas que, una vez combinados, producen $k \times l$ biclusters con mínimo residuo cuadrático medio. Este algoritmo asume una estructura en tablero de ajedrez, por lo que no permite solapamiento entre biclusters.

Otros métodos (SAMBA [298], Kluger *et al*. [333]) están basados en teoría de grafos. Estos algoritmos proponen modelar la matriz de expresión como un grafo bipartito cuyas dos partes se corresponden con genes y muestras, respectivamente. Estos métodos particionan el grafo utilizando técnicas estadísticas [298] o espectrales [333], buscando particiones pesadas del grafo, que se corresponden con biclusters. En SAMBA, un arco entre el gen i y la muestra j se pondera conforme la significación estadística del cambio en el nivel de expresión del gen i en la muestra j. En su lugar, Kluger *et al*. ponderan directamente los arcos con el nivel de expresión a_{ij}, asociado al gen i y muestra j, considerando, por tanto, un grafo bipartido totalmente conectado. El método propuesto por Kluger *et al*. está relacionado con un método anterior de Dhillon de co-clustering de palabras y documentos [89].

Existe una gran variedad de métodos para la identificación de biclusters (para una revisión completa, ver [210]). Algunos de ellos se describen brevemente a continuación:

- Sheng et al. [283] presentan el problema del biclustering en el marco de las Redes Bayesianas, modelando un bicluster como un patrón de frecuencias y

utilizando el muestreo de Gibbs para la estimación de los parámetros. Sólo permite detectar biclusters exclusivos con valores constantes en filas o columnas [210]. Recientenmente, Joshi *et al*. [162], proponen un modelo alternativo de co-clustering de genes y condiciones utilizando muestreos de Gibbs y asumiendo que la distribución de los valores de expresión de los genes y muestras de un cluster siguen distribuciones Gaussianas, permitiendo solapamiento entre los agrupamientos resultantes.

- Getz et al. [118] definen el algoritmo *Couple Two-Way Clustering* que utiliza un clustering jerárquico repetidamente sobre filas y columnas de la matriz, utilizando los clusters de filas resultantes como atributos que se consideran en el clustering de columnas y viceversa. Por lo tanto, este algoritmo no busca patrones que involucren simultáneamente a genes y muestras, sino que efectúa un clustering de filas y luego de columnas (y viceversa).

- Lazzeroni y Owen [195] introducen el modelo PLAID donde la matriz de expresión se describe como la suma de *capas* (cada capa es descrita por un modelo aditivo) que se corresponden con biclusters. Este modelo puede verse como la generalización de un modelo aditivo, ya que tiene en cuenta las interacciones entre biclusters para describir cada elemento a_{ij} de la matriz de expresión. Sólo detecta un bicluster en cada ejecución. Recientemente han surgido algunos algoritmos basados en el modelo PLAID que introducen ciertas mejoras: [313], [314].

- Liu y Wang [203] proponen un algoritmo de eficiencia polinomial que encuentra el bicluster cuadrado (mismo número de genes que de muestras) óptimo que maximiza una medida de similitud.

- Prelic *et al*.[251] comparan la bondad de distintos algoritmos de biclustering y proponen un método divide-y-vencerás de bajo consumo en tiempo llamado *Bimax*.

- Bryan et al. [58] describen un método estocástico de búsqueda de biclusters de mínimo MSR basado en Enfriamiento Simulado (*Simulated Annealing*).

- Pascual-Montano *et al*. [242, 221] implementan una plataforma, denominada bioNMF, que permite aplicar la factorización de matrices (*non-negative matrix factorization*, NMF) al análisis de distintos tipos de datos biológicos, entre ellos, al análisis de matrices de expresión para identificar biclusters.

- Gan *et al*. [114] proponen una interpretación novedosa al problema del biclustering, modelando la identificación de biclusters como la detección de geometrias lineales (hiperplanos) en un espacio de mayor dimensionalidad. Esta aproximación permite detectar simultáneamente biclusters que se rigen por distintos modelos lineales, como el aditivo o el multiplicativo, mediante la definición de restricciones geométricas para los hiperplanos.
- Reiss *et al*. [262] proponen la utilización de información adicional a las matrices de expresión para detectar biclusters de genes corregulados. En particular, Reiss *et al*. [262] definen *cMonkey*, un algoritmo para agrupar genes y muestras en biclusters basado en la similitud de la expresión de los genes, la co-ocurrencia de motivos de regulación en las regiones promotoras de los genes y la existencia de relaciones entre los mismos en redes metabólicas y de regulación genética.
- Bhattacharya y De [49] proponen un algoritmo de biclustering basado en un coeficiente de correlación (*bicorrelation clustering algorithm*, BCCA), y validan los biclusters obtenidos desde un punto de vista biológico examinando la existencia de factores de transcripción comunes en los genes de un bicluster.

Respecto a la utilización de Algortimos Evolutivos para la identificación de biclusters con valores coherentes, el primer trabajo en esta línea fue [53], que extiende el marco propuesto por Cheng y Church en [69] para encontar δ-biclusters. También Aguilar y Divina ([20], [94]) han trabajado en la identificación, utilizando Algoritmos Genéticos, de δ-biclusters maximales de máxima varianza para los genes. Para ello definen una medida de *fitness* que combina el residuo cuadrático medio y la varianza por filas del bicluster. El principal inconveniente es que esta medida requiere el establecimiento de un umbral δ que determine qué biclusers son suficientemente coherentes para ser considerados por el algoritmo. El establecimiento de este umbral, como ya se ha mencionado anteriormente, no es trivial, y requiere cierta información previa que depende del *dataset* concreto que se analice [303].

Los Algoritmos Evolutivos Multiobjetivo (*Multi-objective Evolutionary Algorithms*, MOEA) también han sido recientemente propuestos para la identificación de biclusters en matrices de expresión genética [225, 95]. Los algoritmos MOEA generan un conjunto de soluciones pareto-optimales [83] que optimizan simultáneamente dos o más objetivos que están en conflicto, como el volumen y la varianza o residuo de los genes de los biclusters. Algunos de los algoritmos MOEA más populares son NSGA-II [84], PAES [184] y SPEA2 [349]. Los trabajos [225, 95] utilizan distintos MOEAs para identificar biclusters en matrices de expresión genética. De manera similar, el

trabajo propuesto en [202], propone la utilización de algoritmos de optimización basados en inteligencia de enjambres (*particle swarn optimization*, PSO) [168] para la identificación de biclusters de mínimo residuo y máxima varianza.

2.4 Algoritmo de partida: Gene Shaving.

2.4.1 Objetivos.

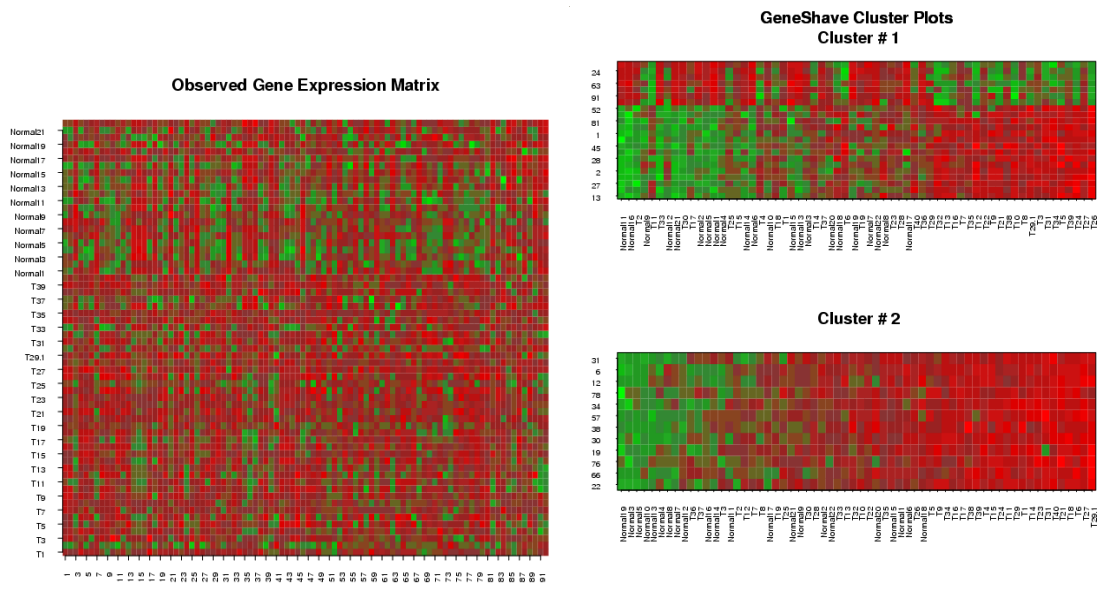
Como la mayor parte de algoritmos de clustering aplicados al análisis de microarrays, el algoritmo Gene Shaving ([134], [135]) ha sido diseñado para identificar grupos de genes coexpresados, esto es, genes que responden de manera similar en las diferentes condiciones en estudio. El hecho de que los genes actúen conjuntamente ante distintas circunstancias resulta un indicio de que estos genes desempeñan una misma función biológica. Por tanto, los grupos de genes identificados se corresponderían, presumiblemente, con módulos funcionales. Además, al tratarse de un algoritmo de clustering no exclusivo, resulta más adecuado para el análisis de matrices de expresión genética porque permite modelar más fielmente los sistemas biológicos reales, en los que un mismo gen puede desempeñar distintas funciones biológicas (pertenecer a distintos clusters).

La particularidad que presenta Gene Shaving, respecto al resto de algoritmos de clustering que se han aplicado al análisis de microarrays, es que permite identificar conjuntos de genes (filas de la matriz de expresión) con patrones de expresión correlacionados y que presenten una alta variación en sus valores a lo largo de todas las muestras o experimentos (columnas de la matriz de expresión). Con este nuevo criterio se pretende conseguir que el nivel de expresión de los genes de un grupo discrimine unas muestras de otras para facilitar la interpretación biológica de los resultados.

Para comprobar visualmente lo que supone la satisfacción de estos criterios, la Figura 2.3 muestra dos clusters obtenidos al aplicar Gene Shaving sobre una matriz de expresión con 92 genes y 62 muestras. Se puede observar la alta correlación entre los genes de cada cluster y la alta varianza que presentan para las distintas condiciones en estudio.

2.4.2 Descripción del algoritmo.

El algoritmo Gene Shaving toma como entrada una matriz de expresión $A_{n\times m}$ con $n \times m$ valores reales, donde las filas representan genes, las columnas muestras, y a_{ij}



(a) **(b)**

Figura 2.3: *(a) Matriz de expresión "alontop.tsv" de 92 genes y 62 muestras tomada de [96]. (b) Clusters obtenidos aplicando Gene Shaving a la matriz anterior.*

representa el nivel de expresión del gen i en la muestra j. El algoritmo también toma como entrada el número de clusters que se desean obtener: M.

Notemos como S_k un cluster de k genes y

$$\overline{a_{S_k}} = (\frac{1}{k}\sum_{i \in S_k} a_{i1}, \frac{1}{k}\sum_{i \in S_k} a_{i2}, ..., \frac{1}{k}\sum_{i \in S_k} a_{im}) \tag{2.2}$$

las medias de las m columnas de la matriz de expresión para los genes del cluster S_k.

Gene Shaving identifica los M clusters secuencialmente. En cada una de las M iteraciones del algoritmo, el objetivo será encontrar un cluster S_k que presente varianza máxima para la media de las columnas del microarray, esto es: $\arg\max var(\overline{a_{S_k}})$.

Para obtener este cluster, Gene Shaving genera una secuencia anidada de clusters:

$$S_n \supset ... \supset S_{k_i} \supset S_{k_j} \supset ... \supset S_1 \tag{2.3}$$

de tamaño decreciente, empezando por $k = n$, el número total de genes, y finalizando con $k = 1$ gen. En cada uno de estos pasos se calcula la 1^a Componente Principal (CP) de cada cluster de genes. Este *eigen-gen* es la combinación lineal de genes con máxima varianza para las muestras. A continuación se descarta una fracción ($\alpha \in [0,1]$) de

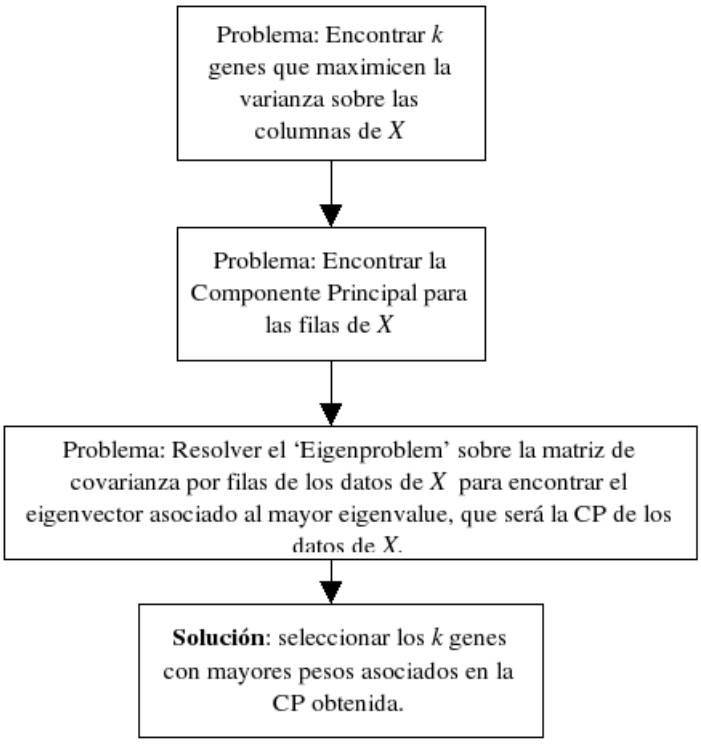


Figura 2.4: *Esquema de la resolución del problema de encontrar k genes que maximicen la varianza para las muestras de A.*

los genes que tienen menor correlación (menor valor absoluto del producto escalar) con este *eigen-gen*, obteniendo así el siguiente cluster de la secuencia (ver Figura 2.4). El proceso se repite hasta que obtenemos un cluster con un único gen.

Cuando se completa la generación de esta secuencia anidada de clusters, el algoritmo selecciona uno de los clusters de la secuencia mediante el cálculo, para cada cluster S_k, de las siguientes medidas de la varianza tomadas de ANOVA (*ANalysis Of VAriance*):

$$\begin{aligned} WithinVariance: &\quad V_W = \frac{1}{m}\sum_{j=1}^{m}[\frac{1}{k}\sum_{i\in S_k}(a_{ij}-\overline{a}_j)^2] \\ BetweenVariance: &\quad V_B = \frac{1}{m}\sum_{j=1}^{m}(\overline{a}_j-\overline{a})^2 \\ TotalVariance: &\quad V_T = \frac{1}{k\times m}\sum_{i\in S_k}\sum_{j=1}^{m}(a_{ij}-\overline{a})^2 = V_W + V_B \end{aligned} \tag{2.4}$$

Donde $\overline{a}_j = \frac{1}{k}\sum_{i\in S_k} a_{ij}$ en las expresiones anteriores.

La *Within Variance* (V_W) mide la variabilidad entre los genes del cluster (cohesión del cluster), por lo que desearemos minimizar esta medida para obtener clusters que

contengan genes con perfiles de expresión similares. La *Between Variance* (V_B) es la varianza del gen promedio del cluster a lo largo de las distintas muestras, por lo que desearemos maximizar esta medida para obtener clusters con alta varianza para las muestras.

Para tomar en cuenta simultáneamente las medidas anteriores, se propone utilizar lo que en ANOVA se denomina porcentaje de varianza explicado por el modelo, o valor R^2:

$$R^2 = 100\frac{V_B}{V_T} = \frac{\frac{V_B}{V_W}}{1+\frac{V_B}{V_W}} \tag{2.5}$$

De modo que valores altos de R^2 implican valores altos de V_B y bajos para V_W. Para conocer si un valor de R^2 para un cluster dado S_k es mayor de lo que cabría esperar por azar, si las filas y columnas de A fueran independientes, se propone la medida denominada GAP.

Sea D_k el valor de R^2 para el cluster S_k, y sea A^{*b} una matriz de datos permutada, obtenida al permutar aleatoriamente los elementos de cada fila de A. Si formamos B matrices de esta manera, definimos la función GAP como:

$$GAP(S_k) = D_k - \overline{D}_k^* \tag{2.6}$$

donde $\overline{D}_k^*$ es el valor promedio de R^2 para S_k en las B matrices permutadas aleatoriamente: $A^{*1}, ..., A^{*B}$. De este modo, un valor alto de GAP revelará un patrón significativo en los datos.

A continuación se selecciona el cluster de la secuencia representada en la ecuación2.3, que maximiza la función GAP.

Después de seleccionar un cluster de la secuencia, la matriz de expresión A se ortogonaliza respecto a la media del cluster seleccionado, promoviendo el descubrimiento de nuevos clusters en las siguientes iteraciones del algoritmo.

El algoritmo completo se muestra a continuación:

Algoritmo Gene Shaving

INPUT: Matriz de expresión A con los valores de cada fila centrados sobre su media y número de clusters M.

1. Calcular el componente principal de las filas de A.
2. Eliminar una proporción α (típicamente 10 %) de las filas que tengan menor producto escalar con el componente principal obtenido.

3. Repetir los pasos 1 y 2 hasta que sólo quede un gen.
   Esto produce una secuencia de clusters de genes anidados:

   $$S_n \supset ... \supset S_{k_i} \supset S_{k_j} \supset ... \supset S_1$$

   donde S_{k_i} representa un cluster de k_i genes.
4. Estimar el tamaño óptimo de cluster (k'), obteniendo un cluster definitivo de genes: $S_{k'}$.
5. Ortogonalizar cada fila de A respecto al gen promedio del cluster $S_{k'}$.
6. Repetir los pasos 1-5, partiendo de la matriz A obtenida en el paso 5, para encontrar el siguiente cluster. Continuar el proceso hasta obtener M clusters.

2.5 Limitaciones del algoritmo Gene Shaving.

Aunque el método Gene Shaving ha sido ampliamente utilizado y referenciado en la literatura, y proporciona buenos resultados, presenta algunas limitaciones que pueden deducirse tras un análisis detenido de su formulación teórica. Algunas de las limitaciones que hemos detectado son las siguientes:

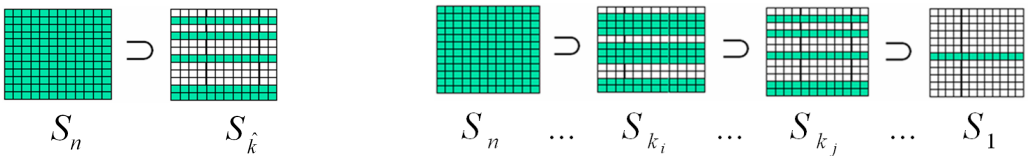
- En cada iteración para obtener la secuencia anidada de clusters de la ecuación 2.3, se seleccionan los genes más correlados con el *eigen-gen* o 1ª CP de las filas de la matriz A. Algunos estudios señalan que el análisis de componentes principales puede ser muy sensible al ruido inherente a las matrices de expresión genética, proponiendo técnicas como el Análisis de Componentes Principales Robusto para solventar esta limitación [147]. Asimismo, se baraja para su futura aplicación el cálculo de Componentes Principales utilizando Kernels (KernelPCA) [277], que no asume linealidad en la expresión de los datos respecto a las CPs.
- Gene Shaving utiliza un criterio para realizar la selección de genes ligeramente distinto al criterio último que se desea maximizar. El proceso de selección de genes se realiza en base al cálculo de la Componente Principal de las filas de A, que proporciona los genes de máxima varianza. De este modo, los clusters de la secuencia anidada resultante presentan un gen promedio con varianza máxima entre muestras. Sin embargo, no sólo interesa maximizar la varianza entre muestras, sino también la cohesión del cluster (similitud entre genes), y este criterio no se utiliza directamente en este proceso de selección iterativo.

Sólo se utiliza al final del proceso, una vez generada la secuencia de clusters de máxima varianza, cuando se calcula el estadístico GAP para cada uno de los clusters de la secuencia para elegir el mejor de todos ellos.

Para abordar esta limitación, proponemos una familia de algoritmos de clustering, basados en Algoritmo Evolutivos, que emplean directamente la métrica GAP (que recoge directamente los dos criterios de interés: máxima varianza de los genes y máxima similitud entre ellos) como medida de *fitness* para obtener buenos agrupamientos. Estos algoritmos se describen en la sección 2.6.

- Gene Shaving agrupa conjuntamente genes que tienen niveles de expresión similares a lo largo de las condiciones (genes coexpresados). Los genes pertenecientes a un mismo cluster responden de forma similar ante distintas circunstancias y condiciones, por lo que probablemente compartirán una misma función biológica. Gene Shaving, además, es un algoritmo de clustering no exclusivo, por lo que permite capturar la realidad biológica de que un mismo gen puede desempeñar distintas funciones. Sin embargo, al tratarse de un algoritmo de clustering, Gene Shaving presenta una limitación intrínseca a este tipo de algoritmos: los genes son agrupados de acuerdo a su comportamiento para todas las condiciones.

  Esto supone una limitación importante al analizar cierto tipo de matrices de expresión genética, particularmente aquellas que consideran numerosas condiciones experimentales y muy heterogéneas. Como se introdujo en la Sección 2.3, el Biclustering ha sido propuesto para abordar esta limitación. En nuestro caso, extenderemos el modelo y la técnica propuesta por Gene Shaving para obtener biclusters, proceso que se describe en la Sección 2.7.



(a) *Single-step FSS* **(b)** *Multiple-step FSS*

Figura 2.5: *Dos esquemas de implementación de la selección de genes utilizando Algoritmos Evolutivos.*

2.6 Aplicación de algoritmos evolutivos a la obtención de clusters de máxima varianza

El proceso de 'Shaving' o selección de genes, implementado por el algoritmo Gene Shaving, puede verse como un proceso de selección de características (o *Feature Subset Selection*, FSS) en múltiples pasos: dado un conjunto de genes S_k con $k \in [2, n]$, buscamos seleccionar un subconjunto con $k \times (1 - \alpha)$ genes: $S_{k\times(1-\alpha)} \subset S_k$, que maximice un criterio dado. Una vez encontrado el subconjunto de genes, se repite el proceso hasta que sólo quede un gen.

En Gene Shaving, el criterio que se optimiza durante la generación de la secuencia anidada de clusters es la varianza entre muestras del gen promedio de los clusters obtenidos. En esta sección, proponemos utilizar directamente el valor GAP para implementar un proceso de selección de características (genes) iterativo utilizando Algoritmos Evolutivos, y en particular, Algoritmos Genéticos (*Genetic Algorithms*, GA) y Algoritmos de Estimación de Distribuciones de Probabilidad (*Estimation of Distribution Algorithms*, EDA), cuyo desempeño en problemas de optimización complejos ha sido sobradamente constatado en la literatura.

Podemos aplicar los Algoritmos Evolutivos (AE) para seleccionar genes siguiendo dos esquemas (ver Figura 2.5):

1. Selección de características en un sólo paso (*Single-step FSS*). Realizar una única ejecución de un AE que tome como entrada la matriz de expresión completa y obtenga como resultado un cluster con alto GAP.

2. Selección de características en múltiples pasos (*Multiple-step FSS*). Generar una secuencia anidada de clusters: $S_n \supset ... \supset S_{k_i} \supset S_{k_j} \supset ... \supset S_1$ de tamaño decreciente desde n hasta 1, descartando paso a paso una fracción de los genes restantes por medio de un AE guiado por la función GAP.

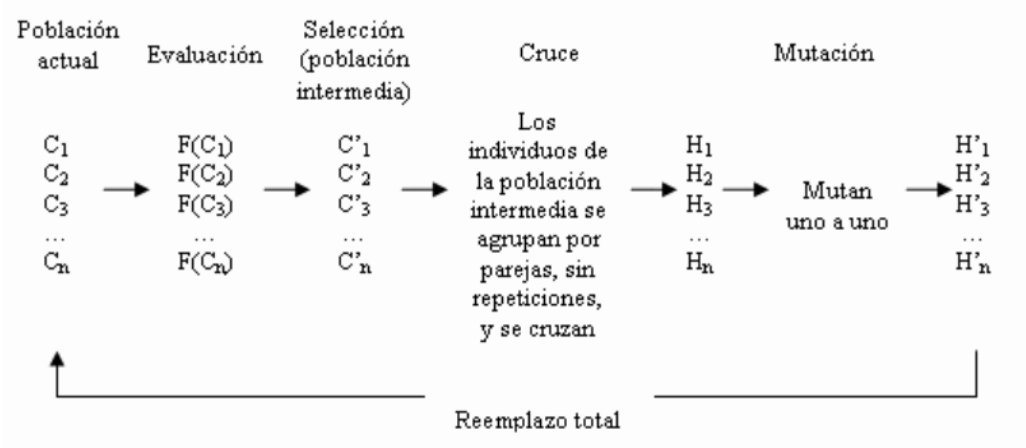


Figura 2.6: *Esquema de seguimiento de la población en un Algoritmo Genético Generacional.*

2.6.1 Clustering utilizando Algoritmos Genéticos: GA-Clustering

Los Algoritmos Genéticos (*Genetic Algorithms*, GA), inicialmente introducidos por Holland [143], son algoritmos de optimización heurística basados en los mecanismos de selección natural y recombinación genética. Los algoritmos genéticos típicamente mantienen una población de tamaño constante de individuos que representan soluciones del espacio de búsqueda. A cada individuo se le asigna un valor de calidad (denominado habitualmente *fitness*) en un proceso de evaluación. Los nuevos individuos se generan recombinando el *material genético* de individuos con buen *fitness*, con lo que la nueva generación conservará muchas de las características de sus "padres". Esto produce que la población vaya mejorando su *fitness* para la función global a optimizar.

Para abordar el problema de seleccionar un grupo de genes que maximice el GAP a partir de una matriz de expresión, se ha implementado un AG generacional con elitismo (ver figura 2.6) con las siguientes características:

- Representación de las soluciones: cada individuo es un string binario de longitud k representando si cada uno de los genes es seleccionado para el cluster o no.
- Selección: muestreo estocástico de Baker. Método de ruleta en el que los *slots* se redimensionan de acuerdo al *fitness* de cada individuo.
- Operador de Cruce: dados dos padres, el cruce mantiene los valores comunes de ambos padres y sortea el resto.
- Operador de Mutación: operador BitFlip. Dado un individuo de la población, una fracción al azar de sus bits se cambian a sus valores complementarios.

- *Fitness*: función GAP.
- Estrategia de reiniciación de la población. Copiamos el mejor individuo a la nueva población y además un 20 % de la nueva población se obtendrá de mutar el mejor individuo de la actual. Aplicamos la reiniciación cuando se consuman un 10 % de las generaciones totales sin cambios en el mejor individuo de la población.

Se ha implementado este algoritmo siguiendo los esquemas *single-step* y *multiple-step* propuestos. Sin embargo, la mayor dimensionalidad del espacio de búsqueda para el primer esquema hace que el AG no converja a soluciones de calidad en un tiempo razonable. El segundo enfoque sí proporciona buenos resultados como se muestra en la sección 2.8. Los mejores resultados son obtenidos utilizando los operadores anteriores y los siguientes parámetros de configuración: Tamaño de la poblacion: $k \cdot \alpha/15 + 20$; Condición de parada : $k \cdot \alpha \cdot 12 + 500$ llamadas a la función de *fitness*; $P_{cruce} = 0{,}9$; $P_{mutacion} = 0{,}2$; donde k es el número total de genes y α la fracción de genes que se eliminan, siendo típicamente $\alpha = 0{,}1$.

2.6.2 Clustering utilizando EDAs: EDA-Clustering

Los Algoritmos de Estimación de Distribuciones de Probabilidad (*Estimation of Distribution Algorithms*, EDA) son Algoritmos Evolutivos caracterizados por la utilización de modelos explícitos de probabilidad para recuperar la información de los individuos seleccionados y para muestrear nuevas soluciones [191]. Los algoritmos EDA han sido extensamente utilizados en bioinformática (para una revisión de EDAs y sus aplicaciones en bioinformática, consultar [31]). En los EDAs, no hay operadores de cruce ni de mutación. En su lugar, un modelo probabilístico se infiere a partir de los individuos seleccionados de la población actual, y la nueva población se genera a partir de la distribución de probabilidad estimada (ver Figura 2.7). Sin embargo, la inferencia de la distribución de probabilidad (modelo probabilístico) no es sencilla, y se hace necesario llegar a un compromiso entre la precisión del modelo y el coste computacional asociado al aprendizaje del mismo.

El modo más sencillo de calcular la distribución de probabilidad consiste en considerar que las variables del problema son independientes. Entonces, la distribución de probabilidad conjunta se convierte en el producto de las distribuciones marginales de las n variables:

$$p_l(\overrightarrow{a}) = \prod_{i=1}^{n} p_l(a_i) \tag{2.7}$$

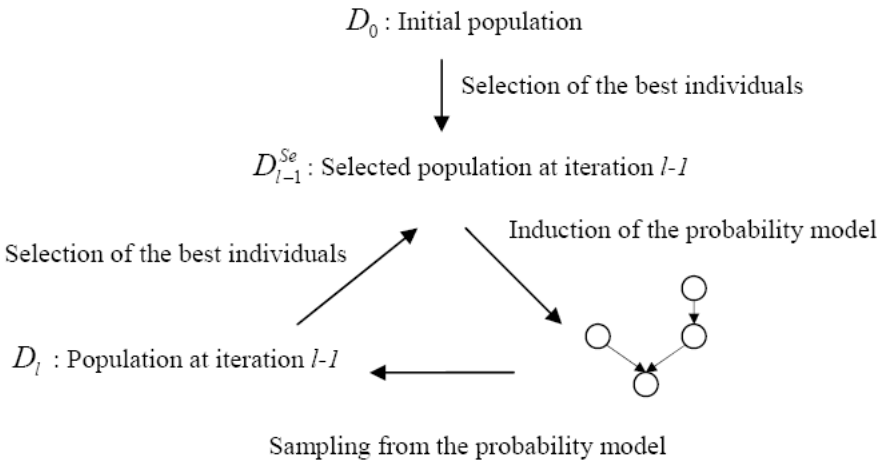


Figura 2.7: *Esquema general de un algoritmo EDA.*

donde $p_l(\overrightarrow{a})$ representa la probabilidad conjunta de los individuos seleccionados en la generación l, $\overrightarrow{a} = (a_1, ..., a_n)$ representa las n variables, y $p_l(a_i)$ son las distribuciones marginales independientes univariantes, que se estiman de las frecuencias marginales:

$$p_l(a_i) = p(a_i \mid D_{l-1}^{Se}) \tag{2.8}$$

donde D_{l-1}^{Se} es el conjunto de los individuos seleccionados en la generación previa: $(l-1)$.

Debido a que se aborda un problema de selección de características, la distribución univariable $p_l(a_i)$ puede calcularse como la fracción de individuos de D_{l-1}^{Se} para los cuales la característica a_i está seleccionada. Este algoritmo fue introducido por [232] y se denomina *Univariate Marginal Distribution Algorithm* (UMDA).

Se ha implementado el algoritmo UMDA y aplicado para solucionar el problema de la selección de genes con los dos esquemas propuestos en la sección anterior: *single-step FSS* y *multiple-step FSS*, obteniendo buenos resultados con las dos aproximaciones (ver sección 2.8). En particular, los mejores resultados se han obtenido con los siguientes parámetros de configuración de los EDA: selección por muestreo estocástico de Baker; $|D_{l-1}^{Se}| = |D_l|/2$; $|D_l| = k \cdot \alpha/15 + 20$ y $\sharp Iterations = 150$ para la selección en mútiples pasos; y $|D_l| = 200$ y $\sharp Iterations = 200$ para la selección en un sólo paso. k es el número de genes de partida y α el porcentaje de genes que se eliminan, con $\alpha = 0{,}1$.

2.7 Biclustering para identificar patrones de máxima varianza

Como ya mencionamos al analizar las limitaciones de Gene Shaving, el clustering sólo permite encontrar agrupamientos de genes que presenten un comportamiento similar para todas las condiciones en estudio. Esto supone una limitación importante al analizar microarrays de expresión, ya que típicamente tendremos decenas de condiciones experimentales, diferentes y heterogéneas, en una misma matriz de expresión. Por tanto, será habitual encontrar grupos de genes que responden de manera similar ante determinadas circunstancias, pero de forma independiente para otras. Para solventar esta limitación, el biclustering ha sido recientemente propuesto para el análisis de matrices de expresión genética.

Esta sección presenta dos líneas de trabajo para la identificación de biclusters en matrices de expresión genética que se ajustan al modelo de cluster definido por Hastie *et al.* para Gene Shaving, esto es, genes que se comportan de manera similar en subconjuntos de condiciones, a la vez que presentan máxima varianza para las mismas. Las propuestas acometidas son:

- Biclustering utilizando Análisis de Componentes Principales, siguiendo el marco propuesto en [135]: Gene&Sample Shaving.
- Biclustering basado en Algoritmos Evolutivos, y en particular, Algoritmos de Estimación de Distribuciones de Probabilidad (EDA): EDA-Biclustering.

2.7.1 Biclustering utilizando Componentes Principales: Gene&Sample Shaving.

En esta sección presentamos el algoritmo Gene&Sample Shaving, esto es: selección de genes y muestras basado en el cálculo de Componentes Principales. El objetivo del algoritmo es extender el marco propuesto por Hastie *et al.*[135] para la obtención de bicluster de máxima coherencia y máxima varianza.

La idea básica que subyace bajo este algoritmo es utilizar el Análisis de Componentes Principales no sólo para obtener los genes de máxima varianza entre muestras (como el algoritmo Gene Shaving), sino también para obtener las muestras que presentan mínima varianza para un conjunto dado de genes. De este modo, al igual que se eliminan los genes *menos correlados* con la 1ª Componente Principal *de las filas de A* (en un proceso denominado 'Gene Shaving'), también se eliminan las muestras *más*

correladas con la 1ª Componente Principal *de las columnas de A* (en un proceso que, por analogía, puede denominarse 'Sample Shaving'). Es decir, se eliminan los genes de mínima varianza para las muestras del bicluster, y las muestras de máxima varianza para los genes del bicluster. De este modo, se obtienen biclusters en los que los genes presentan un comportamiento muy similar para una misma muestra, pero de máxima varianza entre las muestras del bicluster.

Esta eliminación de genes y muestras de la matriz de expresión puede producirse siguiendo distintos esquemas:

- Esquema 'Gene Shaving' + 'Sample Shaving'. Consiste en aplicar primero el proceso de selección de genes implementado en el algoritmo 'Gene Shaving' y posteriormente un proceso iterativo de selección de columnas o 'Sample Shaving'. Este proceso consiste en eliminar las columnas más correladas con la 1ª CP de las columnas, refinando los resultados obtenidos por 'Gene Shaving'. De este modo, se producen biclusters que mejoran el GAP del cluster original. El proceso 'Sample Shaving' puede aplicarse sobre el cluster final devuelto por 'Gene Shaving' o incluso sobre todos los clusters $S_n, ..., S_k, ..., S_1$ de la secuencia:$S_n \supset ... \supset S_k \supset S_1$ obtenida con 'Gene Shaving', devolviendo en cualquiera de los casos el bicluster obtenido de mayor GAP.

- Esquema 'Sample Shaving' + 'Gene Shaving'. Consiste en aplicar primero el proceso iterativo 'Sample Shaving' para obtener clusters de columnas que presenten mínima varianza en su expresión para todos los genes, y posteriormente refinar estos clusters de columnas eliminando genes mediante 'Gene Shaving' para encontrar biclusters que mejoren en GAP al cluster de partida. Sin embargo, este esquema no resulta conveniente para el análisis de matrices de expresión. Al disponer típicamente de miles de genes en las matrices de expresión, no existe ningún grupo de muestras que presente valores similares de expresión para todos los genes en estudio, por lo que comenzar el proceso aplicando 'Sample Shaving' sobre toda la matriz de expresión no proporciona buenos resultados.

- Esquema que combina 'Gene Shaving' con 'Sample Shaving'. Otra opción que consideramos consiste en intercalar los pasos de las dos estrategias: la eliminación de los genes menos correlados con la 1ª CP para las filas con la eliminación de las columnas más correladas con la 1ª CP de las columnas. Sin embargo este esquema presenta las mismas limitaciones que el 'Sample Shaving'+'Gene Shaving', porque en los primeros pasos de 'Sample Shaving' apenas se habrán eliminado genes de la matriz de expresión. De este modo, al inicio del proceso

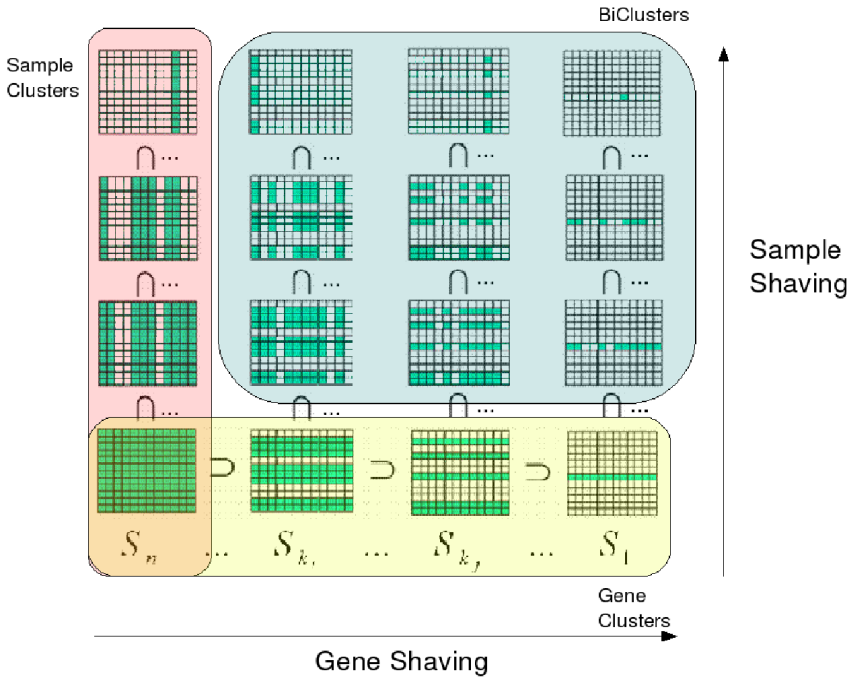


Figura 2.8: *Esquema del algoritmo Gene&Sample Shaving.*

se eliminarán columnas en base a sus niveles de expresión para casi todos los genes de la matriz, lo que resulta inadecuado para obtener buenos biclusters.

Dado que las matrices de expresión presentan muchos más genes que muestras (típicamente miles de genes y decenas de muestras), los mejores resultados se consiguen siguiendo la variante 'Gene Shaving' + 'Sample Shaving'. La Figura 2.8 muestra un esquema del funcionamiento de este algoritmo: primero se aplica 'Gene Shaving' para obtener una secuencia anidada de clusters y, a continuación, se aplica 'Sample Shaving' sobre cada uno de los clusters de esta secuencia para refinarlos y obtener los biclusters correspondientes. El bicluster de mejor GAP es finalmente elegido, y la matriz de expresión se ortogonaliza respecto a la media de este bicluster, promoviendo la identificación de señales novedosas en siguientes iteraciones del algoritmo, y a la vez posibilitando el solapamiento entre biclusters.

2.7.2 Biclustering utilizando EDAs: EDA-Biclustering.

En esta sección se presenta el algoritmo EDA-Biclustering, que implementa una selección de genes y muestras basado en un algoritmo EDA y guiado por completo por el estadístico GAP. Este algoritmo utiliza el EDA tipo UMDA, presentado en la

Algorithm 1 Pseudocódigo de una algoritmo EDA

1. Generar M individuos (la población inicial) de forma aleatoria para construir D_{l-1}.
2. Repetir hasta que se alcance el criterio de parada.
   a) Seleccionar $N <= M$ individuos de D_{l-1}de acuerdo con algún método de selección.
   b) Estimar la distribución de probabilidad $p_l(x) = p(x|D_{l-1})$a partir de los individuos seleccionados.
   c) Muestrear M individuos (la nueva población) a partir de $p_l(x)$ para construir D_l.

Figura 2.9: *Algoritmo general EDA.*

sección 2.6.2 (ver figura2.9), para obtener biclusters de máximo GAP, es decir, de máxima coherencia y máxima varianza. De este modo, se extiende el marco propuesto por el algoritmo EDA-Clustering para obtener biclusters en lugar de clusters.

Para ello, sólo es necesario que incluir en la codificación de las soluciones, los m bits necesarios para representar las m muestras de la matriz de expresión. De este modo, una solución (un bicluster) queda representada como una cadena de $n + m$ bits, los n primeros asociados a cada uno de los genes y los m últimos asociados a cada una de las muestras, con 1 indicando que el gen o la muestra está incluido en el bicluster, y 0 indicando que no lo está.

Hemos implementado el algoritmo EDA-Biclustering siguiendo el esquema *single-step FSS* descrito anteriormente, obteniendo los mejores resultados para $|D_l| = 300$, $\sharp Iterations = 300$ y utilizando los mismos operadores que en EDA-Clustering. Los resultados se muestran en la Sección 2.8.2.

2.8 Experimentos y Análisis de Resultados.

En esta sección se muestran los resultados obtenidos por el algoritmo de partida, Gene Shaving, y los distintos algoritmos de clustering y biclustering propuestos a lo largo del capítulo. Los algoritmos de clustering son evaluados sobre los datos de expresión del ciclo celular de la levadura (*S. cerevisiae*) obtenidos de [71]. Este *dataset* contiene los niveles de expresión de 2879 genes en 17 muestras que cubren, aproximadamente, dos ciclos celulares completos de la levadura. Los datos han sido seleccionados y preprocesados conforme a [69]. Nos referiremos a estos datos como *dataset de la levadura*. Los algoritmos de biclustering se evalúan, además de sobre el *dataset* anterior, sobre un *dataset* humano que recoge los niveles de expresión de 4026 genes bajo 96 muestras tisulares humanas, que se dividen en 9 tipos distintos de linfomas y tejidos sanos. Este *dataset* se ha obtenido de [25] y su preprocesamiento ha consistido en la eliminación de genes duplicados, la sustitución de valores perdidos mediante el cálculo del valor promedio de los 10 vecinos más cercanos y el centrado en 0 de las filas de la matriz resultante. Nos referiremos al mismo como *dataset de linfomas*.

Empleamos el *dataset* de linfomas para evaluar los algoritmos de biclustering porque recoge un mayor número de condiciones experimentales y una mayor heterogeneidad en las mismas: mientras que la matriz de la levadura contiene una serie temporal del ciclo celular, el *dataset* de linfomas recoge 9 tipos distintos de muestras de cáncer y tejidos sanos humanos, por lo que es posible que existan genes que se comporten de manera similar sólo para un subconjunto de las condiciones, es decir, sólo para cierto/s tipo/s de cáncer.

La evaluación comparativa entre los distintos algoritmos de clustering y biclustering se centra en el valor GAP y el tamaño de los clusers y biclusters encontrados.

Validación e interpretación biológica de los resultados

La interpretación biológica de los resultados consiste, fundamentalmente, en determinar qué funciones biológicas y caminos metabólicos están significativamente asociados a un conjunto de genes.

El proyecto *Gene Ontology* (GO) [32] tiene como objetivo construir ontologías en las que se asocian (o *anotan*) los productos de los genes a los procesos biológicos en los que participan (ontología *biological process*), sus funciones moleculares (ontología *molecular function*) y los componentes celulares donde actúan (en la ontología *cellular component*). Las anotaciones de G0 y otros recursos de anotaciones funcionales, como GenMAPP [270] y KEGG [165], han sido ampliamente utilizadas para

la identificación de módulos funcionales significativamente representados en grupos de genes [98, 23, 341, 144, 338]. Estas herramientas emplean distintos test estadísticos para determinar si una determinada ruta metabólica o proceso biológico están sobre-representados en una lista de genes. Esta aproximación se basa en la asunción de que un grupo de genes co-expresados, puesto que exhiben el mismo comportamiento en distintas muestras o para distintas condiciones experimentales, compartirá una misma función biológica [321]. Aunque esta asunción puede considerarse cierta como observación general, trabajos recientes han mostrado que la definición de clase funcional de este tipo de recursos (GO, KEGG, etc.) no implica necesariamente la co-expresión de los genes de la clase [213, 230].

En nuestro trabajo, buscamos los términos de la ontología *Gene Ontology* más significativamente representados por los genes de cada uno de los clusters y biclusters obtenidos, esto es, determinaremos si algún término GO está presente en los genes de un cierto grupo con una probabilidad mayor que de la esperada, calculando el correspondiente *p-value* utilizando la distribución hipergeométrica y la corrección de Bonferroni para múltiples hipótesis. Para obtener los términos de GO significativamente sobre-representados en un conjunto de genes, utilizamos *GO Term Finder* [56]. En particular, nos centramos en la ontología *biological process* de GO, ya que asumimos que la co-expresión de un grupo de genes indica potencialmente que éstos comparten una misma función biológica, y por tanto participan en un mismo *proceso biológico* (mientras que pueden tener distinta *función molecular* y actuar en distinto *compartimento celular*).

Como se muestra a continuación, la asociación del proceso biológico más representativo (el de menor p-value) a cada grupo de genes obtenido, nos permite validar los resultados obtenidos. Dado que los clusters y biclusters pueden asignarse con fiabilidad a procesos biológicos de GO, puede concluirse que los resultados de los algoritmos propuestos se ajustan a la clasificación funcional conocida (en este caso, la proporcionada por *Gene Ontology*) y por tanto, son fiables para extraer nuevo conocimiento biológico. Del mismo modo, la asignación de procesos biológicos a los grupos de genes puede ser empleada para asignar función a genes para los que hasta ahora se desconocía su papel: si uno de estos genes está fuertemente co-expresado con un grupo de genes que presentan una función biológica conocida, ese gen probablemente desempeñe la misma función.

2.8.1 Algoritmos de Clustering.

2.8.1.1 Comparativa de Resultados.

La Tabla 2.1 muestra el GAP y tamaño promedio para 100 clusters obtenidos en 10 ejecuciones de cada uno de los algoritmos: Gene Shaving, GA-Clustering y EDA-Clustering (con esquema de selección en múltiples pasos y en un solo paso).

Algorithm	No. genes	GAP
Gene Shaving	13.26 (10.33)	61.89 (23.87)
GA-Clustering	14.56 (4.01)	79.92 (3.8)
EDA-Clustering (multiple-step)	15.3 (6.4)	81.87 (4.8)
EDA-Clustering (single-step)	35.53 (10.1)	72.64 (4.6)

Cuadro 2.1: *Valores medios y desviaciones típicas (entre paréntesis) del GAP y tamaño para 100 clusters.*

Los resultados de la Tabla 2.1 muestran que los algoritmos GA-Clustering y EDA-Clustering (con esquemas de selección *single-step* y *multiple-step*) obtienen clusters de mayor GAP y tamaño que Gene Shaving. Aplicando un *t-test* de dos colas, comprobamos que las mejoras en términos de GAP son estadísticamente significativas ($p - valor < 0{,}05$) para GA-Clustering y EDA-Clustering (*multiple-step*) respecto a los resultados obtenidos por Gene Shaving. No podemos constatar que la mejora en términos de GAP para el EDA-Clustering (*single-step*) sea significativa, sin embargo este algoritmo obtiene muy buenos resultados en GAP y los clusters de mayor tamaño. Además, los resultados en términos de tamaño de los clusters para este algoritmo son significativamente mejores (t-test) que los obtenidos por cualquier otro método de clustering ($p - valor < 0{,}05$). La Figura 2.10 muestra el valor GAP y tamaño de un subconjunto de soluciones obtenidas por cada algoritmo.

2.8.1.2 Interpretación biológica.

Como se ha mencionado anteriormente, utilizamos la información de la ontología *Gene Ontology* para determinar el proceso biológico de GO más significativo asociado a los genes de cada uno de los clusters obtenidos. Los patrones biológicos más relevantes son identificados cuando consideramos clusters con alto valor GAP y bajo p-valor para su función más representativa (ver Figura 2.11). De este modo, es posible validar los algoritmos propuestos e interpretar los resultados para extraer conocimiento

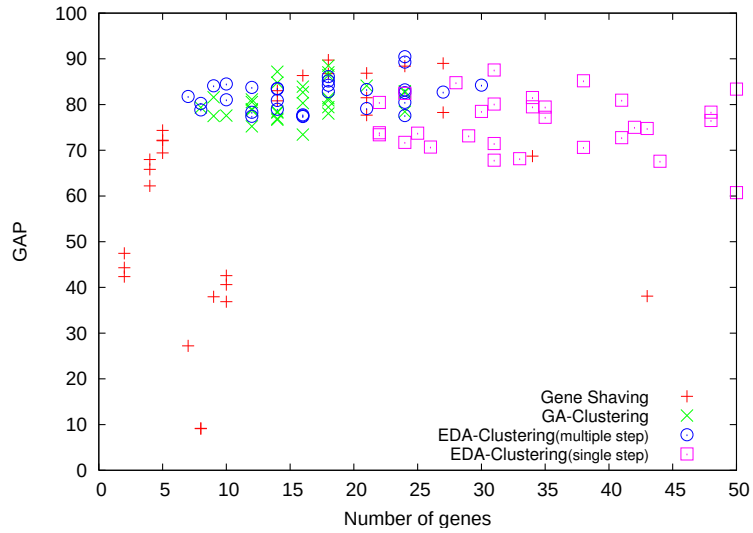
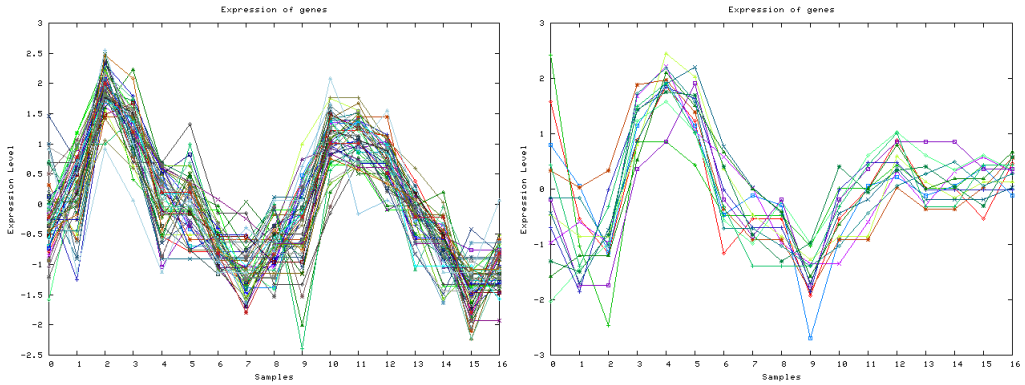


Figura 2.10: *Diagrama de dispersión del GAP y tamaño (número de genes) de los clusters obtenidos con cada algoritmo de clustering en el dataset de la levadura. Sólo los resultados de las tres primeras ejecuciones (30 clusters) se muestran para cada algoritmo.*

biológico novedoso de forma fiable. Por ejemplo, observando el primer gráfico de expresión de la Figura 2.11 podemos comprobar la correspondencia entre el proceso biológico 'Metabolismo de ADN', que es el de menor p-value asociado a este cluster, y el comportamiento de los genes que pertenecen a dicho cluster, que se sobre-expresan en las muestras 2-3 y 10-12, muestras asociadas a la *fase S* del ciclo celular, en la que tiene lugar la duplicación de ADN [71].

Un ejemplo que ilustra la capacidad de EDA-Clustering (esquema *single-step*) para obtener clusters de mayor tamaño que todos los demás métodos y de buena calidad, puede observarse en la Figura 2.12. Todos los algoritmos han encontrado un cluster significativamente asociado a la duplicación de ADN. Podemos comprobar que aunque Gene-Shaving, GA-Clustering y EDA-Clustering *mutiple-step* presentan valores más altos de GAP para sus clusters, EDA-Clustering *single-step* agrupa muchos más genes en el cluster (18 genes en el cluster obtenido por Gene-Shaving frente a 38 en el obtenido por EDA-Clustering) con patrones de expresión muy similares, buen valor GAP y p-valor muy bajo. Entonces, el cluster encontrado por EDA-Clustering *single-step* parece ser el más valioso desde un punto de vista biológico.

La figura 2.13 muestra otro ejemplo. Los dos clusters están significativamente asociados al proceso *DNA unwinding* (desenrrollamiento de ADN) pero el obtenido por EDA-Clustering tiene mayor GAP y mayor tamaño que el obtenido por Gene-Shaving.



(a) **(b)**

Figura 2.11: *Perfiles de expresión para algunos clusters biológicamente significativos, encontrados con EDA-Clustering. El eje de abcisas representa las muestras de la matriz de expresión, y el de ordenadas los niveles de expresión. Cada línea del gráfico representa un gen perteneciente al cluster. (a) Single-step EDA-Clustering. DNA metabolism. P-value:* 18×10^{-13}. *GAP:*83,38. *size:50 genes. (b) Multiple-step EDA-Clustering. Sulfur metabolism. P-value:* $7{,}2 \times 10^{-15}$. *GAP:*83,4. *size:*14 *genes.*

2.8.2 Algoritmos de biclustering.

2.8.2.1 Comparativa de resultados.

Dataset de la levadura.

Los algoritmos de biclustering implementados (ver tabla 2.2) mejoran los resultados de los algoritmos de clustering de partida. Gene&Sample Shaving obtiene biclusters con mejor GAP que Gene Shaving ($p-valor < 0{,}01$), por lo que obtiene patrones más refinados y de mejor calidad. EDA-Biclustering también mejora las dos versiones de EDA-Clustering en términos de GAP ($p-valores < 0{,}01$). De hecho, EDA-Biclustering muestra los mejores resultados para este *dataset* de todos los algoritmos propuestos.

Dataset de linfomas.

La tabla 2.3 muestra el GAP y tamaño promedio para 500 biclusters obtenidos en 10 ejecuciones de cada uno de los algoritmos: Gene&Sample Shaving, EDA Biclustering y también el algoritmo de clustering de partida: Gene Shaving, que ha sido ejecutado como referencia en la comparativa con los algoritmos de biclustering.

De nuevo, comprobamos que los algoritmos de biclustering obtienen patrones con mayor GAP que Gene Shaving, siendo similar el promedio del número de genes, y bastante menor el número de columnas. A raíz de estos resultados podemos afirmar que los algoritmos de biclustering propuestos mejoran significativamente

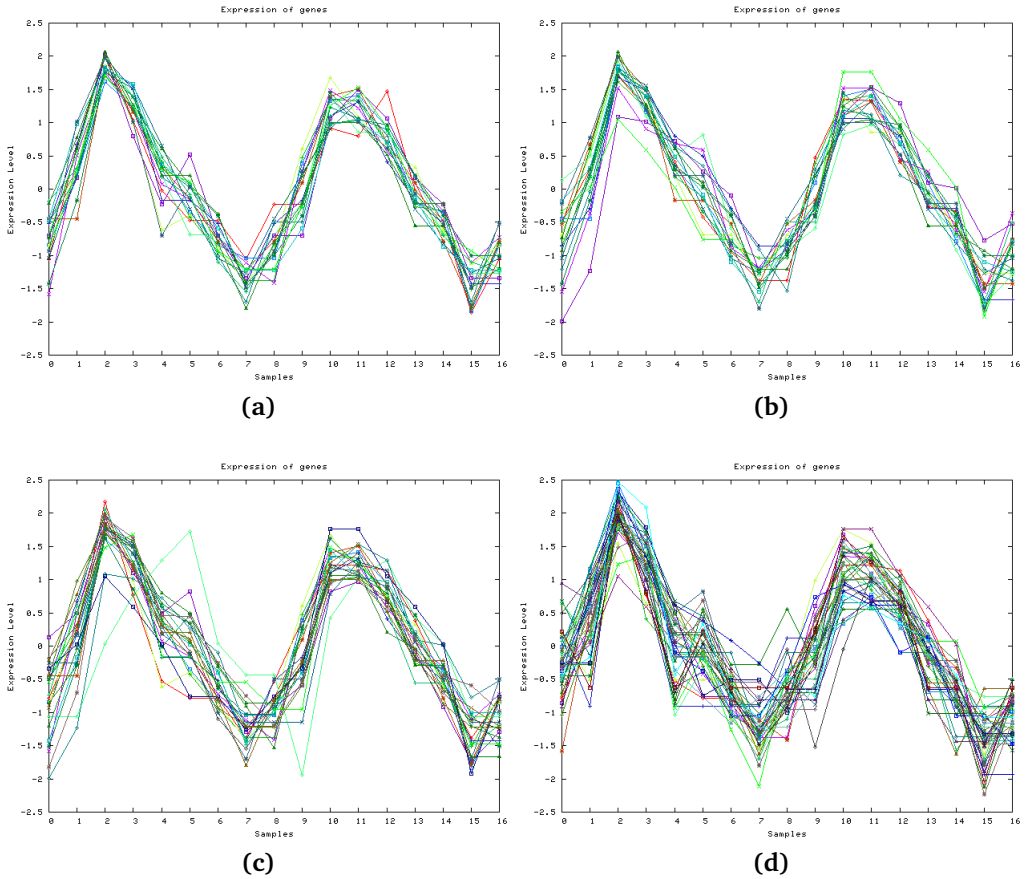
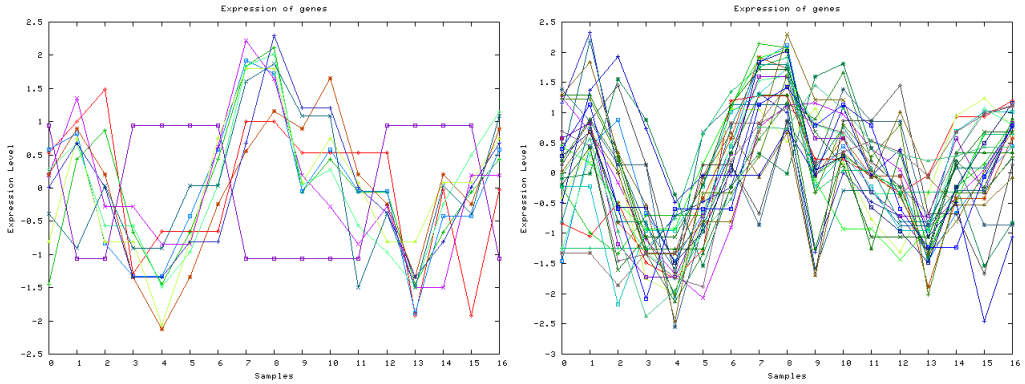


Figura 2.12: *Perfiles de expresión genética para clusters significativamente asociados a DNA replication obtenidos con Gene-Shaving, GA-Clustering, multiple-step EDA-Clustering y single-step EDA-Clustering. (a) Gene-Shaving. P-value:* $3{,}7 \times 10^{-09}$. *GAP:* 89,73. *size:*18 *genes (b) GA-Clustering. P-value:* $4{,}2 \times 10^{-09}$. *GAP:* 87,1. *size:*18 *genes (c) Multiple-step EDA-Clustering. P-value:* $1{,}5 \times 10^{-10}$. *GAP:* 89,27. *size:*24 *genes (d) Single-step EDA-Clustering. P-value:*$9{,}3 \times 10^{-14}$. *GAP:*85,2. *size:*38 *genes.*

($p - valor < 0{,}05$) los resultados de Gene Shaving en términos de GAP. Además, la baja dispersión asociada al valor GAP de los resultados de los nuevos algoritmos de biclustering propuestos confirma la robustez y bondad de los mismos.

El clustering ofrece peores resultados en este *dataset* por el alto número de condiciones y la alta heterogeneidad de las mismas, que hacen que muy pocos genes exhiban el mismo comportamiento para todas las condiciones. El biclustering resulta más adecuado para el análisis de estos datos, pues permite encontrar patrones de más calidad que involucran sólo las muestras para las que los genes del bicluster presentan un comportamiento muy similar entre sí y a la vez de máxima varianza entre las muestras.



(a) **(b)**

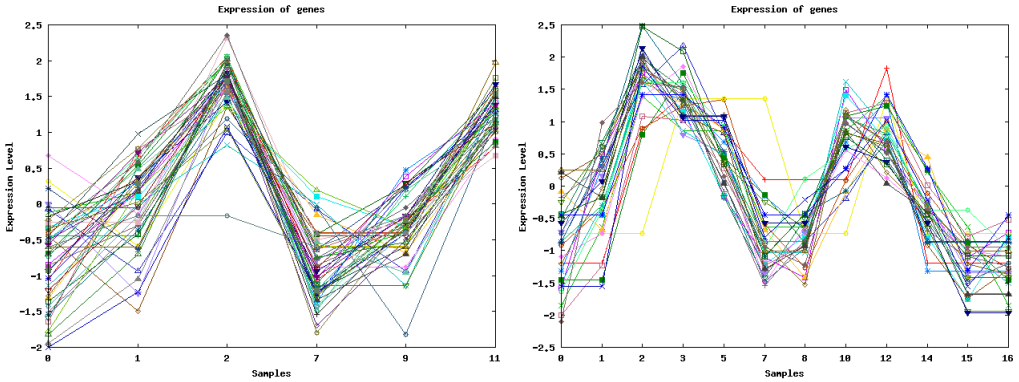
Figura 2.13: *Perfiles de expresión genética para clusters signficativamente asociados a DNA unwinding obtenidos con Gene-Shaving y single-step EDA-Clustering. (a)Gene-Shaving. P-value:* $4{,}6 \times 10^{-11}$. *GAP:*42,58. *size:*10 *genes (b)Single-step EDA-Clustering. P-value:* $1{,}64 \times 10^{-05}$. *GAP:* 73. *size:*29 *genes*

Cuadro 2.2: *Dataset de la levadura. Valores medios y desviaciones típicas (entre paréntesis) del GAP y tamaño para los biclusters obtenidos.*

Algorithm	No. genes	No. cols	GAP
Gene&Sample Shaving	11.53 (7.5)	4.66 (2.7)	86.6 (8.4)
EDA Biclustering	25.1 (4.4)	6.7 (2.6)	88.79 (4.1)

Cuadro 2.3: *Dataset de linfomas. Valores medios y desviaciones típicas (entre paréntesis) del GAP y tamaño de los agrupamientos obtenidos.*

Algorithm	No. genes	No. cols	GAP
Gene Shaving	13.28 (96.6)	96	52.13 (17.3)
Gene&Sample Shaving	10.98 (7.3)	14.89 (14.2)	83.99 (6.9)
EDA Biclustering	20.24 (6.6)	17.92 (4.5)	68.56 (8.3)



(a) **(b)**

Figura 2.14: *(a) Perfiles de expresión genética para bicluster significativamente asociado a DNA metabolism obtenido con Gene&Sample Shaving. P-value:* 2×10^{-10}. *GAP:* 92,9 . *size:* 48 *genes,* 6 *condiciones.(b) Perfiles de expresión genética para bicluster significativamente asociado a mitotic cell cycle obtenidos con EDA Biclustering. P-value:* $5{,}35 \times 10^{-5}$. *GAP:* 86,32 .*size:* 30 *genes,* 12 *condiciones.*

2.8.2.2 Interpretación biológica.

Dataset de la levadura.

En la Figura 2.14 mostramos los perfiles de expresión de dos biclusters obtenidos con Gene&Sample Shaving y EDA Biclustering cuyos genes están significativamente asociados al *DNA metabolism* y *mitotic cell cycle*. Podemos comprobar que los biclusters recogen patrones en los que los genes se comportan de manera muy similar y, además, la variabilidad en el comportamiento de los genes difiere mucho entre unas columnas y otras.

Dataset de linfomas.

Los genes humanos del *dataset* de linfomas son poco conocidos respecto a su función biológica, comparados con los genes de la levadura del anterior *dataset*: 2601 genes de los 2879 genes del *dataset* de la levadura tienen al menos un proceso biológico conocido en GO frente a los 2228 de los 4026 genes humanos del *dataset* de linfomas. Este menor conocimiento de la funcionalidad de los genes humanos hace que la significación biológica de los agrupamientos obtenidos en este caso sea menor que con el *dataset* anterior.

Sin embargo, en este *dataset* contamos con distintos tipos de condiciones experimentales presentes en la matriz de expresión y conocidas *a priori*: *Follicular Lymphoma (FL)*, *Diffuse Large B-Cell Lymphoma (DLBCL)*, *Chronic lymphocytic leukaemia*

(CLL), *Germinal centre B cell (GCB)*, *Activated Blood B (Act Blood B)*, *Resting Blood B (Rest. Blood B)*, *Resting/Activated T (Rest/Act T)*, *LymphNode/Tonsil* y *Transformed Cell Lines*. Los tres primeros tipos se corresponden con tejidos cancerígenos, y el resto con distintos tipos de tejidos y células sanas. Por lo tanto, podemos determinar si los biclusters obtenidos agrupan columnas significativamente asociadas a ciertos tipos de condiciones, de manera similar a como anteriormente procedimos con los genes y sus funciones biológicas.

Para determinar si nuestros biclusters se ajustan a la clasificación de condiciones disponible, calcularemos la significación estadística (p-value) de cada tipo de condición en cada bicluster, corrigiendo de nuevo los p-values obtenidos con la corrección de Bonferroni para múltiples hipótesis. Los resultados obtenidos son prometedores. Por ejemplo, la Figura 2.15 muestra los niveles de expresión de los genes de un bicluster obtenido con EDA Biclustering. Este resultado es destacable porque el bicluster recoge significativamente las condiciones asociadas al tipo *Chronic lymphocytic leukaemia (CLL)* con un p-value corregido de 1.4e-05, y además los genes del bicluster muestran valores de expresión bajos (están subexpresados) en las condiciones asociadas a este tipo de cáncer (columnas numeradas del 83 al 94), y valores de expresión altos (están sobreexpresados) en un popurrí del resto de condiciones, en el que hay representación de todos los tipos presentes en la matriz de expresión. Por lo tanto, el comportamiento de los genes recogidos en este bicluster discrimina el cáncer CLL del resto de tejidos sanos y cancerígenos considerados, por lo que puede investigarse la función biológica de estos genes para extraer alguna conclusión del estudio.

2.9 Conclusiones.

Este capítulo se ha centrado en el desarrollo de técnicas no supervisadas para el análisis y la extracción de conocimiento de matrices de expresión genética obtenidas de microarrays. En particular, esta línea de trabajo tiene su origen en el algoritmo Gene Shaving, uno de los métodos de clustering más ampliamente extendidos y referenciados en el análisis de microarrays. Su principal aportación reside en la búsqueda de grupos de genes con perfiles de expresión similares que además presenten varianza máxima para las muestras. Esto permite identificar grupos funcionales que se comportan de forma muy diferente para distintos tipos de muestras, abriendo una puerta para la caracterización de dichos tipos y el análisis detallado de los procesos biológicos subyacentes que pueden ser responsables de las diferencias de comportamiento.

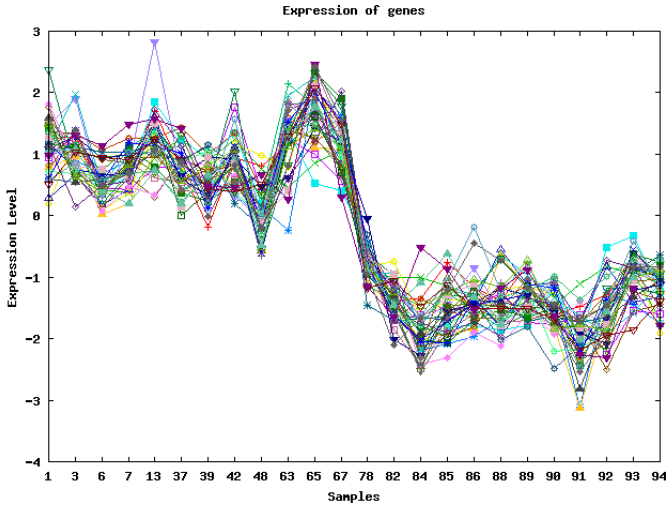


Figura 2.15: *Perfiles de expresión para los genes contenidos en un bicluster obtenido con EDA Biclustering sobre los datos de linfoma humano. El bicluster recoge 10 de las 11 columnas asociadas al cáncer CLL (p-value corregido para CLL:* $1{,}4 \times 10^{-5}$*), mostrando valores bajos de expresión en sus genes para las condiciones asociadas a este tipo (columnas 84 a 94), frente al resto. GAP:* 90,22 *.size:* 39 *genes,* 24 *condiciones.*

Nuestro trabajo toma la definición de cluster de Gene Shaving y presenta nuevos algoritmos de Clustering y Biclustering basados en el Análisis de Componentes Principales y Algoritmos Evolutivos (Algoritmos Genéticos y Algoritmos de Estimación de Distribuciones de Probabilidad). Estos métodos extienden el algoritmo Gene Shaving en el tipo de patrones que pueden identificar (no solo clusters sino también biclusters) y en la calidad de los mismos. Los algoritmos propuestos han sido los siguientes:

- GA-Clustering: clustering utilizando Algoritmos Genéticos.
- EDA-Clustering: clustering utilizando Algoritmos EDA.
- Gene&Sample Shaving: biclustering utilizando Análisis de Componentes Principales.
- EDA-Biclustering: biclustering utilizando Algoritmos EDA.

Los resultados experimentales sobre *datasets* de ciclo celular de la levadura y de linfomas humanos demuestran que los algoritmos de clustering y biclustering propuestos mejoran los resultados de Gene-Shaving en términos de calidad (valor GAP) y de tamaño de los clusters/biclusters obtenidos. Para destacar la potencialidad del biclustering, también se han presentado evaluaciones sobre el *dataset* de linfoma humano, que recoge un mayor número de condiciones y más heterogéneas. Los biclusters obtenidos en este *dataset* mejoran ampliamente los resultados de Gene Shaving,

encontrando patrones más refinados, que incluso permiten discriminar ciertos tipos tumorales del resto de muestras. Este capítulo muestra la metodología utilizada para la validación e interpretación biológica de los resultados, utilizando anotaciones de Gene Ontology y la clasificación de condiciones proporcionada por Alizadeh *et al.* [25] para el *dataset* de linfomas.

Capítulo

3

Biclustering espectral posibilístico en matrices de expresión genética

3.1 Motivación y objetivos.

El Biclustering permite identificar conjuntos de genes (muestras) que exhiben un comportamiento similar para un subconjunto de las muestras (genes) de las matrices de expresión, lo que lo convierte en una herramienta muy extendida para el análisis de matrices de expresión. Los algoritmos de clásicos de agrupamiento, que obtienen agrupaciones exclusivas (sin solapamiento), no permiten capturar la realidad biológica de que un gen puede participar en distintos procesos biológicos y desempeñar más de una función. Los algoritmos de agrupamiento basados en tecnología difusa recogen de forma natural la posibilidad de que un gen pueda desempeñar simultáneamente distintas funciones biológicas (pertenecer a varios grupos), por lo que resultan apropiados para el análisis de este tipo de datos.

La Sección 1.2.3 revisa algunos criterios habituales para la detección de biclusters. Uno de los criterios más extendidos es el Residuo Cuadrático Medio o *Mean Squared Residue* (MSR), basado en la asunción de que los datos de expresión se representan conforme al modelo aditivo (Sección 2.3). Aunque existen distintos métodos que tienen como objetivo la identificación de δ-biclusters (biclusters con MSR inferior a un valor δ establecido *a priori*) [69, 324, 323, 334], estos algoritmos presentan ciertas limitaciones. Primero, al ser algoritmos greedy iterativos (todos excepto [323]), pueden converger en un óptimo local y dejar sin identificar biclusters importantes [346]. Segundo, el establecimiento de un umbral δ adecuado no es trivial y requiere de cierto conocimiento a priori que depende del *dataset* [303]. Dhillon [70] propone un

algoritmo greedy para minimizar el MSR que presenta el inconveniente añadido de que asume una estructura en tablero de ajedrez, por lo que no permite solapamiento entre biclusters.

Por otro lado, los principios de clustering espectral han sido satisfactoriamente aplicados en otros campos como el co-clustering de palabras y documentos [89]. Estos métodos se basan en el modelado de matrices de datos como un grafo bipartito, cuyas dos partes se corresponden con los elementos fila y los elementos columna, respectivamente, y en la partición de este grafo en subgrafos (submatrices) utilizando principios de la teoría espectral de grafos, es decir, utilizando las propiedades del espectro de la matriz laplaciana del grafo [89]. Estas técnicas han sido posteriormente aplicadas al biclustering de matrices de expresión genética por Kluger *et al.* [333]. Sin embargo, los algoritmos [333, 89] presentan limitaciones importantes cuando son aplicados al análisis de matrices de expresión genéticas, puesto que permiten identificar únicamente estructuras en forma de tablero de ajedrez [333] y biclusters exclusivos [89], respectivamente.

En este capítulo presentamos un nuevo método de biclustering basado en tecnología difusa y técnicas espectrales de clustering. Este nuevo método, que hemos denominado Biclustering Espectral Posibilístico (*Possibilistic Spectral Biclustering*, PSB), utiliza el Residuo Cuadrático Medio [69] (*Mean Squared Residue*, MSR) como medida de calidad y permite identificar biclusters potencialmente solapados en matrices de expresión genética.

El algoritmo PSB utiliza principios de clustering espectral [333, 89], pero busca simultáneamente k biclusters en la matriz de expresión que pueden estar posicionados arbitrariamente y ser potencialmente solapados. De esta forma, el algoritmo PSB permite capturar la realidad biológica de que un gen puede presentar distintas funciones biológicas, actuando conjuntamente con distintos grupos de genes. Además, el PSB utiliza el residuo cuadrático medio como medida para seleccionar el mejor bicluster en lugar de buscar δ-biclusters con un valor *a priori* de δ (como [69], [324] y [334]).

El capítulo se estructura como sigue. Las sección 3.2 repasa los trabajos previos en técnicas difusas de clustering y biclustering aplicados al análisis de matrices de expresión genética. Para una revisión detallada de otras técnicas de clustering y biclustering, nos remitimos a las secciones 2.2 y 2.3 del capítulo 2. En la Sección 3.3, resumimos las propuestas de Dhillon [89] y Kluger et al. [333] para obtener biclusters utilizando principios del clustering espectral, señalando los inconvenientes que surgen al aplicar estos métodos sobre matrices de expresión genética (3.4). En la Sección 3.5 presentamos el algoritmo de Biclustering Espectral Posibilístico (PSB) como

solución a estos problemas. La Sección 3.6 presenta los resultados experimentales, que incluyen una evaluación comparativa con otros métodos de biclustering en datos sintéticos y dos *datasets* reales y la validación de los biclusters obtenidos con las anotaciones de Gene Ontology [32]. Finalmente, la Sección 3.7 presenta las conclusiones de esta línea de trabajo.

3.2 Aplicación del clustering difuso al análisis de microarrays

El clustering difuso ha sido aplicado al análisis de matrices de expresión genética por su capacidad para asignar un gen a más de un grupo simultáneamente. Ésto permite un modelado más fiel de la realidad biológica, en la que un gen puede participar en distintos procesos biológicos.

El algoritmo K-medias Difuso (*Fuzzy C-means*, FCM), es la extensión difusa del algoritmo K-medias de clustering y basa el cálculo del grado de pertenencia de un objeto a un grupo en la distancia de ese objeto al centroide que representa a dicho grupo [47, 86]. Este algoritmo es descrito con detalle en la Sección 1.2.2.1.

Numerosas variantes de FCM han sido propuestas en los últimos años para el análisis de matrices de expresión genética, incluyendo una variante heurística que incorpora el Análisis de Componentes Principales (PCA) y el clustering jerárquico [115]; el denominado *Fuzzy J-Means*, que aplica una búsqueda variable en el entorno de la solución para evitar óptimos locales [44]; métodos que combinan el clustering difuso con técnicas de enfriamiento simulado y redes neuronales [215] o con máquinas de vector soporte (Support Vector Machines, SVM) para mejorar su rendimiento [233, 216]. También se ha propuesto un enfoque que propone el uso de FCM alineando los centroides en un mapa de Kohonen (SOM) [241]. El algoritmo FLAME propone la asignación de los grados de pertenencia de un objeto a los distintos grupos en base a los valores de pertenencia asociados a los objetos vecinos más cernanos [111]. [33] propone proyectar los datos de la matriz de expresión original en varios espacios de menor dimensión y aplicar FCM sobre los mismos, combinando los clusters obtenidos.

Otra familia de métodos de clustering difuso es la que emplea Modelos Mixtos Gaussianos (*Gaussian Mixture Model*, GMM) [131, 256, 336] para modelar los datos de expresión y estimar las funciones de pertenencia difusas. Sin embargo, estos métodos asumen que los datos de expresión se distribuyen según una combinación

de distribuciones normales, asunción que no siempre se satisface en datos reales, incluso aunque éstos hayan sido sometidos a distintas tranformaciones para mejorar la normalidad [336].

Recientemente, también han sido propuestos algunos métodos de clustering difuso que incorporan conocimiento *a priori* tomado de otras fuentes de datos, como [300], que incorpora información de Gene Ontology.

3.3 Enfoque de partida: biclustering espectral.

Los trabajos de Dhillon [89] y Kluger [333] han aplicado principios del Clustering Espectral [235] al Biclustering, resultando en lo que hoy denominamos Biclustering Espectral.

La idea principal en la que se sustentan las aproximaciones espectrales al análisis de microarrays es la modelización de la matriz de datos $A_{n\times m}$ como un grafo completo bipartito $G = \{F, C, E\}$, en el que:

- $F = \{f_1, f_2, ..., f_n\}$ es el conjunto de nodos que representa a las filas de A,
- $C = \{c_1, c_2, ..., c_m\}$ es el conjunto de nodos que representa las columnas de A,
- $E = \{\{f_i, c_j\} : f_i \in F, c_j \in C\}$ es el conjunto de arcos que conectan los nodos de F y C.

Los arcos de E solo pueden conectar nodos de F y C, por ser G un grafo bipartito. El peso asociado al arco $\{f_i, c_j\}$ es igual a la entrada (i, j) de la matriz $A_{n\times m}$. La matriz de adyacencia (Y) de G se define como:

$$Y = \begin{pmatrix} 0 & A \\ A^T & 0 \end{pmatrix}$$

Por tanto Y tiene dimensiones $(n+m)\times(n+m)$, donde las n primeras entradas de las filas y las columnas de Y hacen referencia a los nodos de F (filas de A) y las m últimas entradas de las filas y columnas de Y hacen referencia a los nodos de C (columnas de A). De este modo, cada entradas (i, j) de Y contiene el peso del arco que une el nodo i y el nodo j en G. Los ceros indican que no existe asociación entre dos nodos de F ni entre dos nodos de C, por ser G bipartito.

La figura 3.1 muestra un ejemplo de grafo bipartito totalmente conectado obtenido a partir de la matriz de expresión:

$$A = \begin{pmatrix} 1 & 3 & 4 \\ 2 & 3 & 2 \\ 1 & 2 & 4 \end{pmatrix} \tag{3.1}$$

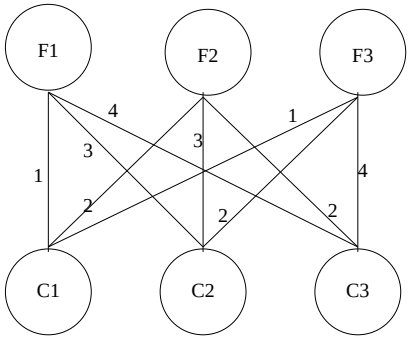


Figura 3.1: *Ejemplo de grafo bipartito que representa la matriz de expresión de la ecuación 3.1.*

con matriz de adyacencia: $Y = \begin{pmatrix} 0 & 0 & 0 & 1 & 3 & 4 \\ 0 & 0 & 0 & 2 & 3 & 2 \\ 0 & 0 & 0 & 1 & 2 & 4 \\ 1 & 2 & 1 & 0 & 0 & 0 \\ 3 & 3 & 2 & 0 & 0 & 0 \\ 4 & 2 & 4 & 0 & 0 & 0 \end{pmatrix}$

Los subgrafos G_i de G: $G_i = \{F_i, C_i, E_i \ : F_i \subset F, C_i \subset C, E_i \subset E\}$ también representan grafos bipartitos compuestos de subconjuntos de filas y columnas de la matriz de expresión A. De este modo, los subgrafos G_i de G son equivalentes a los biclusters $B_i = (F_i, C_i)$ de la matriz de expresión A.

Definición de corte de un grafo.

En teoría de grafos, si notamos como $G = (V, E)$ un grafo simple y V_1, V_2 una partición en dos subconjuntos de V, definimos el **corte** (*cut*) del grafo dado por dicha partición como:

$$cut(V_1, V_2) = \sum_{i \in V_1, j \in V_2} Y_{ij}$$

Es decir, el corte de un grafo es la suma de los pesos de las aristas que conectan vértices de los conjuntos V_1 y V_2. Esta definición puede generalizarse fácilmente al corte para una partición en k subconjuntos $V_1, V_2, ...V_k$:

$$cut(V_1, V_2, ..., V_k) = \sum_{i<j} cut(V_i, V_j)$$

El problema de encontrar la partición del grafo G en k subconjuntos de vértices de forma que se minimice el valor del corte, es un problema clásico de teoría de grafos de complejidad NP-Completo.

En nuestro caso, partir el grafo G maximizando los pesos de los subgrafos G_i que representan los biclusters B_i es equivalente, por tanto, a minimizar el corte (cut) entre los subgrafos [105]. Esta aproximación nos permite buscar genes sobre-expresados para un subconjunto de condiciones en la matriz de expresión A.

En el caso del grafo bipartito G, el problema se reformula en los siguientes términos: buscar una partición $B_1, B_2, ..., B_k$ con $B_i = (F_i, C_i)$ tal que se minimice el corte:

$$cut(B_1, B_2, ..., B_k) = min_{V_1, V_2, ..., V_k} cut(V_1, V_2, ..., V_k) \tag{3.2}$$

para todo $V_1, V_2, ..., V_k$ tal que $\bigcup V_i = V$ y $V_i \cap V_j = \emptyset$.

Sin embargo, existe una solución trivial para el problema de optimización anterior. La partición: $B_1 = (F, C); B_i = \emptyset \; \forall i \neq 1$, en la que uno de los subconjuntos engloba todo el grafo G y el resto están vacios; minimiza el valor del corte (0).

Por consiguiente, es necesario proponer otra formulación que no sólo tenga en cuenta el corte entre los conjuntos de vértices, sino también algún otro factor cuya optimización produzca particiones mejor balanceadas. La idea fundamental consiste en asignar a cada nodo (fila o columna) i un peso $weight(i)$, definiendo la matriz de pesos W como la matriz que contiene en su diagonal el peso de cada nodo:

$$W_{ij} = \begin{cases} weight(i) & i = j \\ 0 & i \neq j \end{cases} \tag{3.3}$$

Con $weight(i) = \sum_{i \in V_l} weight(i) = \sum_{i \in V_l} W_{ii}$

De este modo, la función a minimizar se puede definir ahora como:

$$Q(V_1, V_2) = \frac{cut(V_1, V_2)}{weight(V_1)} + \frac{cut(V_1, V_2)}{weight(V_2)} \tag{3.4}$$

favoreciendo que los subgrafos en que se particiona G tengan pesos parecidos y a la vez el corte sea mínimo.

Existen diversas alternativas para asignar pesos a los nodos de G. Una de estas alternativas consiste en asignar peso 1 a cada nodo [128], obteniéndose la función $Radio - cut$:

$$Radio - cut(V_1, V_2) = \frac{cut(V_1, V_2)}{|V_1|} + \frac{cut(V_1, V_2)}{|V_2|} \tag{3.5}$$

La función $Radio - cut$ favorece la obtención de biclusters con un número similar de nodos.

Otra alternativa consiste en asignar a cada nodo la suma de los pesos de los arcos que inciden en él:

$$weight(i) = \sum_k Y_{ik}$$

lo que hace a W equivalente a la denominada matriz de grado (D):

$$D_{ij} = \begin{cases} \sum_k Y_{ik} & i = j \\ 0 & i \neq j \end{cases} \tag{3.6}$$

Siguiendo con esta alternativa, el peso para un conjunto de nodos (V_1) se calcula, por tanto, como la suma de los pesos de los arcos que inciden en cada nodo del conjunto. Los arcos que inciden en estos nodos pueden tener origen en nodos pertenecientes al mismo conjunto V_1 o de nodos que no pertenezcan a V_1. Considerando una partición de V en dos subconjuntos V_1 y V_2, se define el peso para un conjunto de nodos V_1 como:

$$weight(V_1) = \sum_{i \in V_1} \sum_k A_{ik} = cut(V_1, V_2) + within(V_1)$$

donde $within(V_1)$ es el peso de los arcos que tienen origen y destino en V_1 y $cut(V_1, V_2)$ son, por definición, los arcos fuera de V_1 que conectan este subgrafo con V_2.

A partir de la función general para Q y utilizando este criterio, se obtiene la función $Normalized - cut\ (N)$ [284]:

$$N(V_1, V_2) = \frac{cut(V_1, V_2)}{\sum_{i \in V_1} \sum_k A_{ik}} + \frac{cut(V_1, V_2)}{\sum_{i \in V_2} \sum_k A_{ik}} = 2 - S(V_1, V_2)$$

con:

$$S(V_1, V_2) = \frac{within(V_1)}{weight(V_1)} + \frac{within(V_2)}{weight(V_2)}$$

El uso de la función $Normalized - cut$ para obtener una partición de G en dos subgrafos aparece en el trabajo de Dhillon [89]. Este resultado permite calcular un vector de partición aproximado z para la partición de un grafo G, minimizando la función $Normalized - cut$ [124]. Las primeras n entradas del vector z se corresponden con las n filas de la matriz A, mientras que las siguientes m entradas se corresponden con las columnas de A. El vector z_2 es el eigenvector asociado al menor eigenvalue λ_2 obtenido al resolver el eigenproblema:

$$Lz = \lambda Wz$$

donde L es la matriz Laplaciana de G definida como $L = D - Y$.

Utilizando la definición de matriz Laplaciana y la función $Normalized - cut$ para asignar pesos a los nodos (haciendo por tanto equivalentes las matrices W y D), podemos reformular el eigenproblema anterior como:

$$\begin{pmatrix} D_1 & -A \\ -A^T & D_2 \end{pmatrix} \begin{pmatrix} x \\ y \end{pmatrix} = \lambda \begin{pmatrix} D_1 & 0 \\ 0 & D_2 \end{pmatrix} \begin{pmatrix} x \\ y \end{pmatrix} \tag{3.7}$$

donde

$$\begin{array}{l} D_1(i,i) = \sum_j a_{ij} \quad \text{(suma de los arcos incidentes en la fila i)} \\ D_2(j,j) = \sum_i a_{ij} \quad \text{(suma de los arcos incidentes en la columna j)} \end{array} \tag{3.8}$$

Finalmente obtenemos las ecuaciones:

$$\left.\begin{array}{rcl} D_1^{-\frac{1}{2}} M D_2^{-\frac{1}{2}} v & = & \sigma u \\ D_2^{-\frac{1}{2}} M^T D_1^{-\frac{1}{2}} u & = & \sigma v \end{array}\right\} \tag{3.9}$$

donde:

$$u = D_1^{\frac{1}{2}} x$$

$$v = D_2^{\frac{1}{2}} y$$

$$\sigma = (1 - \lambda)$$

y el segundo eigenvector es:

$$z_2 = \begin{pmatrix} D_1^{-\frac{1}{2}} u_2 \\ D_2^{-\frac{1}{2}} v_2 \end{pmatrix} \tag{3.10}$$

Este problema algebraico puede resolverse utilizando la descomposición en valores singulares (Singular Value Decomposition, SVD)[1] que nos proporciona $min(n, m)$ soluciones ortogonales z.

De este modo, z_2 representa una solución aproximada a la partición de G en dos subgrafos de forma que se minimice la función $Normalized - cut$.

[1]SVD nos permite expresar la matriz $A_n = D_1^{-\frac{1}{2}} M D_2^{-\frac{1}{2}}$ como: $A_n = USV^T$ donde las columnas de U son los eigenvectores izquierdos (en particular, la segunda columna de U corresponde a u_2), las filas de V^T son los eigenvectores derechos (la segunda fila corresponde a v_2) y los eigenvalores se encuentran en la diagonal de S ($\sigma_2 = S_{22}$). Además, el cuadrado de cada eigenvalue σ es proporcional a la cantidad de varianza de A_n explicada por el eigenvector asociado.

Para obtener una partición en más de dos subgrafos se han propuesto distintos enfoques en la literatura: Dhillon [89] propone aplicar un algoritmo de clustering K-medias sobre los primeros l eigenvectores[2] $u_2, u_3, ..., u_l$ y $v_2, v_3, ..., v_l$ con $l = \lceil log_2 K \rceil$ para construir:

$$Z = \begin{pmatrix} D_1^{-\frac{1}{2}} U_2^l \\ D_2^{-\frac{1}{2}} V_2^l \end{pmatrix}$$

Z representa una matriz de $(m+n)$ filas por l columnas donde cada fila i contiene información sobre a cuál de los K biclusters pertenece la fila o columna de la matriz inicial a la que representa dicha entrada i.

Aplicando un algoritmo K-medias en las filas de Z, obtenemos clusters de filas y columnas de la matriz de expresión A, cada uno de los cuales representa un bicluster de A.

3.3.1 Algoritmo de Dhillon.

A continuación se presenta la formulación del algoritmo de Dhillon y sus distintas fases. Las entradas del algoritmo son la matriz de expresión $A_{n \times m}$ y el número de particiones en filas y columnas (o número de biclusters) K.

1. Calcular la matriz de adyacencia Y a partir de la matriz de expresión $A_{n \times m}$. Si utilizamos la función $Normalized - cut$, calcularemos $Y = D_1^{-\frac{1}{2}} M D_2^{-\frac{1}{2}}$.
2. Aplicar SVD para descomponer Y como $Y = USV^T$.
3. Construir $U_2^l = [u_2, u_3, ..., u_l]$ y $V_2^l = [v_2, v_3, ..., v_l]$. Dhillon propone tomar $l = \lceil log_2 K \rceil$ [89]. También se puede elegir l de acuerdo con los valores de los eigenvalores, ya que estos representan la cantidad de variabilidad de la matriz explicada por cada par de eigenvectores.
4. Obtener $Z = \begin{pmatrix} D_1^{-\frac{1}{2}} U_2^l \\ D_2^{-\frac{1}{2}} V_2^l \end{pmatrix}$, donde $Z_{i.}$ con $0 \leq i < n$ representa a la fila f_i y $Z_{j.}$ con $n \leq j < n+m$ representa a la columna c_{j-n}.
5. Aplicar K-medias sobre las filas de Z para obtener K centroides $M = m_1, m_2, ..., m_K$.

[2] El primer eigenvector z_1 no es considerado para el clustering puesto que es constante y, por lo tanto, no aporta información sobre cómo particionar el grafo.

6. Construir K biclusters B_1, B_2, ..., B_K de forma que una fila i de Z, representada por $Z_{i.}$ (que a su vez representa a una fila o columna de A), pertenecerá al bicluster B_j si la distancia de $Z_{i.}$ al centroide $m_j \in M$ es menor que la distancia a cualquier otro centroide $m_l \in M; m_l \neq m_j$.

3.4 Limitaciones de los algoritmos espectrales de biclustering.

El algoritmo de bicluster espectral de Dhillon[89] presenta ciertas limitaciones al ser aplicado sobre datos de expresión genética:

- *Biclusters no solapados*. Utilizando un algoritmo de clustering K-medias para agrupar las filas de Z, cada elemento (que se corresponde con una fila o columna de A, es decir, cada gen o muestra) sólo puede pertenecer a un bicluster, por lo que se obtienen biclusters no solapados. Esto resulta de enorme importancia al analizar datos de expresión genética, puesto que impide detectar distintos patrones que compartan un mismo gen, o lo que resulta equivalente, impide que un gen pueda estar involucrado en distintos procesos biológicos conjuntamente con distintos grupos de genes.
- La función $Normalized-cut$ produce subgrafos en los que se maximiza el peso de los arcos, por lo que, trasladado a las matrices de expresión genética, obtenemos biclusters de sobre-expresión. Sin embargo, también resulta interesante en este ámbito detectar biclusters de sub-expresión, es decir, grupos de genes que están sub-expresados para un subconjunto de muestras.
- El espacio de soluciones es pequeño si solo consideramos una partición de la matriz Z. Ampliando este espacio de búsqueda podrían encontrarse biclusters de mejor calidad.

3.5 Biclustering espectral posibilístico.

A raíz de las limitaciones de los algoritmos de biclustering espectral presentados en la sección anterior, proponemos un nuevo método de biclustering espectral, denominado biclustering espectral posibilístico (PSB) que hace uso del clustering posibilístico para identificar biclusters potencialmente solapados y de mínimo MSR en matrices de expresión genética. En esta sección describimos los distintos componentes del algoritmo PSB.

3.5.1 Tecnología difusa para el clustering de microarrays.

Desde que, en 1965 L. Zadeh propusiera la Teoría de Conjuntos Difusos [340], ésta ha sido extensamente utilizada para modelar y analizar sistemas con ruido e incertidumbre. Como ya se introdujo en la Sección 1.2.2, el clustering difuso ha sido utilizado para el análisis de matrices de expresión genética, ya que permite modelar más fielmente la realidad biológica, al contemplar la posibilidad de que un gen desempeñe varias funciones y participe en distintos procesos biológicos (clusters).

Los algoritmos de clustering difusos probabilísticos ([47, 116]) imponen la restricción de que la suma de los grados de pertenencia de un objeto a los distintos grupos sea 1. Si eliminamos esta restricción, obtenemos lo que se conoce como clustering posibilístico [188].

3.5.1.1 Clustering posibilístico.

La técnica de clustering *k*-medias difuso es adecuada cuando los grados de pertenencia se interpretan como probabilidades o grados de compartición: esto obliga que la suma de los grados de pertenencia de un objeto en los distintos clusters sea 1. Así, por ejemplo, si tenemos definidos las clases de los individuos rubios, morenos y pelirrojos, un individuo puede ser rubio con un 0.5, moreno con un 0.4 y pelirrojo con un 0.1.

Sin embargo, este planteamiento puede no resultar conveniente en otras situaciones. Por ejemplo, en el análisis de matrices de expresión genética, si equiparamos los grupos obtenidos con funciones o procesos biológicos, la participación de un gen en un proceso biológico determinado no debe restringir su participación en otros procesos. En casos como éste, sería deseable la independencia entre los grados de pertenencia de un elemento a distintos grupos.

Un método que permite solucionar este problema es el cluster posibilístico [188], en el que no se impone la condición de que la suma de los grados de pertenencia a los clusters sea 1. En este caso, los grados de pertenencia pueden interpretarse como posibilidad o tipicidad de un objeto en un determinado cluster.

Otro factor a tener en cuenta es que el clustering posibilístico es menos sensible al ruido que el tradicional *k*-medias difuso, ya que los *outliers* pueden tener asociados bajo grado de pertenencia a todos los clusters (con lo que no influyen significativamente en el cálculo de los centroides de estos clusters). Esta mejor respuesta ante datos con ruido hace que el clustering posibilístico resulte más adecuado para el análisis de matrices de expresión.

Retomando la descripción de algoritmos de cluster posibilísticos, la supresión de la restricción de suma 1 para el grado de pertenencia de cada objeto a los distintos clusters conlleva un problema: la optimización de la función objetivo proporciona la solución trivial en la que todos los grados de pertenencia son 0. Como solución, Krishnapuram y Keller [188] proponen la siguiente función objetivo:

$$J_m(L,U) = \sum_{i=1}^{C}\sum_{j=1}^{N}(u_{ij})^m d_{ij}^2 + \sum_{i=1}^{C}\eta_i \sum_{j=1}^{N}(1-u_{ij})^m$$

donde C es el número de clusters que se buscan; N el número de objetos a agrupar; L es la C-tupla de prototipos utilizada para calcular las distancias; U_{CxN} es la matriz de grados de pertenencia, con u_{ij} notando el grado de pertenencia del objeto j al cluster i; d_{ij} es la distancia del centroide i al objeto j; η_i es un factor para cada cluster que se describe a continuación; y m es el parámetro de fuzzificación, que toma valores entre 1 y 2, dependiendo del *dataset* y del método de normalización[178].

Tras derivar la función objetivo se puede observar que el grado de pertenencia de un punto a un cluster difuso sólo depende de la distancia a ese cluster. Obtenemos la siguiente ecuación para el cálculo iterativo de los grados de pertenencia:

$$u_{ij} = \frac{1}{1+(\frac{d_{ij}^2}{\eta_i})^{\frac{1}{m-1}}}$$

El valor η_i se interpreta como la distancia a la que el grado de pertenencia de un punto se hace 0,5. Krishnapuram y Keller [188] proponen un η_i proporcional a la distancia media dentro del cluster i:

$$\eta_i = K\frac{\sum_{j=1}^{N} u_{ij}^m d_{ij}^2}{\sum_{j=1}^{N} u_{ij}^m}$$

donde habitualmente $K = 1$.

De este modo, se propone un esquema de algoritmo en el que inicialmente se aplica k-medias difuso, en base a los resultados obtenidos se calculan los η_i para los distintos clusters y se itera usando el cálculo de grados de pertenencia descrito para el posibilístico hasta que se produce convergencia. Una segunda fase puede recalcular los η_i teniendo en cuenta sólo aquellos puntos que pertenecen en ese momento a cada cluster con un grado de pertenencia superior a uno dado, típicamente 0,4 (esto reduce la influencia del ruido). Se vuelve a iterar con esos nuevos pesos para η_i hasta que se converja. La descripción completa del algoritmo se presenta a continuación:

Algoritmo de clustering posibilístico

1. Calcular U según k-medias difuso
2. Estimar η_i con $\eta_i = K\frac{\sum_{j=1}^{N} u_{ij}^m d_{ij}^2}{\sum_{j=1}^{N} u_{ij}^m}$.
3. Repetir:
   a) Calcular prototipos c_i
   b) Actualizar los grados de pertenencia U

   Hasta que se converja
4. Reestimar η_i siguiendo un α-corte determinado
5. Repetir:
   a) Calcular prototipos c_i
   b) Actualizar grados de pertenencia U

   Hasta que se converja
6. Asumiendo la distancia euclídea, los prototipos o centroides se calculan mediante:

$$c_i = \frac{\sum_{j=1}^{N} u_{ij}^m x_j}{sum_{j=1}^{N} u_{ij}^m}$$

3.5.1.2 Posibilístico mejorado

El clustering posibilístico básico de Krishnapuram y Keller [188] presenta el problema de el algoritmo tiende a producir clusters coincidentes (excesivo solapamiento), aunque existan distintos clusters claramente distinguibles en los datos. Esto se debe a que, al relajar la condición del clustering probabilístico, por la cual los grados de pertenencia suman 1, los distintos clusters ya no tienen que "competir" por los puntos, y esto puede ocasionar que lleguen a coincidir al tender a ocupar la misma zona del espacio de los datos.

Para evitar este problema Zhang y Leung [343] proponen modificar la función objetivo de forma que combine las ideas probabilísticas y posibilísticas. De esta forma, proponen el uso de grados de pertenencia probabilísticos $U^{(f)}$ y grados de pertenencia posibilísticos $U^{(p)}$ con la siguiente función objetivo:

$$J(L, U^{(p)}, U^{(f)}) = \sum_{i=1}^{C}\sum_{k=1}^{N}(u_{ik}^{(f)})^{m_f}((u_{ik}^{(p)})^{m_p} d_{ik}^2 + \eta_i(1 - u_{ik}^{(p)})^{m_p}) =$$

$$= \sum_{i=1}^{C}\sum_{k=1}^{N}(u_{ij}^{(f)})^{m_f}(u_{ij}^{(p)})^{m_p}d_{ik}^2 + \sum_{i=1}^{C}\eta_i\sum_{k=1}^{N}(u_{ik}^{(f)})^{m_f}(1-u_{ik}^{(p)})^{m_p}$$

donde:

- $u_{ik}^{(p)}$ es el grado de pertenencia posibilístico del punto k al cluster i.
- $u_{ik}^{(f)}$ es el grado de pertenencia probabilístico del punto k al cluster i.
- m_p y m_f son los factores de fuzzificación posibilístico y probabilístico respectivamente.

El cálculo de los grados de pertenencia se realiza iterativamente según las siguientes ecuaciones:

$$u_{ik}^{(p)} = \frac{1}{1+(\frac{d_{ik}^2}{\eta_i})^{\frac{1}{m_p-1}}}$$

$$u_{ik}^{(f)} = \frac{1}{\sum_{j=1}^{C}(\frac{(u_{ik}^{(p)})^{\frac{m_p-1}{2}}d_{ik}}{(u_{jk}^{(p)})^{\frac{m_p-1}{2}}d_{jk}})^{\frac{2}{m_f-1}}}$$

Si usamos la distancia euclídea obtenemos la siguiente variante del algoritmo posibilístico, denominada algoritmo posibilístico mejorado (*Improved Possibilistic Clustering*, IPC) [343]:

Algoritmo mejorado de clustering posibilístico (IPC)

1. Utilizando K-medias difuso calcular $U^{(f)}$ (grados de pertenencia probabilísticos) y L (prototipos)
2. Calcular los $\eta_i = \frac{\sum_{k=1}^{N}(u_{ik}^{(f)})^{m_f}d_{ik}^2}{\sum_{k=1}^{N}(u_{ik}^{(f)})^{m_f}}$
3. Repetir:
   a) Reestimar los grados de pertenencia posibilísticos $U^{(p)}$
   b) Reestimar los grados de pertenencia probabilísticos $U^{(f)}$
   c) Calcular los prototipos como $c_i = \frac{\sum_{k=1}^{N}(u_{ik}^{(p)})^{m_p}(u_{ik}^{(f)})^{m_f}x_k}{\sum_{k=1}^{N}(u_{ik}^{(p)})^{m_p}(u_{ik}^{(f)})^{m_f}}$ para la distancia euclídea

   Hasta que se converja

   (opcional)
4. Recalcular los $\eta_i = \frac{\sum_{k=1}^{N}(u_{ik}^{(p)})^{m_p}(u_{ik}^{(f)})^{m_f}d_{ik}^2}{\sum_{k=1}^{N}(u_{ik}^{(p)})^{m_p}(u_{ik}^{(f)})^{m_f}}$
5. Repetir:

a) Reestimar los grados de pertenencia posibilísticos $U^{(p)}$

b) Reestimar los grados de pertenencia probabilísticos $U^{(f)}$

c) Calcular los prototipos como $c_i = \frac{\sum_{k=1}^{N}(u_{ik}^{(p)})^{m_p}(u_{ik}^{(f)})^{m_f} x_k}{\sum_{k=1}^{N}(u_{ik}^{(p)})^{m_p}(u_{ik}^{(f)})^{m_f}}$ para la distancia euclídea

Hasta que se converja

En particular, en lugar de utilizar el algoritmo IPC directamente, se propone la siguiente metodología:

- Aplicar IPC con un número de clusters c mayor que el esperado.
- Eliminar clusters duplicados.
- Aplicar IPC utilizando el resto de prototipos.

3.5.2 Varias matrices de partición.

Como se ha expuesto anteriormente, la utilización de una sóla matriz de partición o espectro Z produce una única partición del grafo G, y por lo tanto, biclusters no solapados. Sin embargo, la selección de diferentes conjuntos de eigenvectores z podría generar varias matrices Z, y obtener distintas particiones del grafo que pudieran ser potencialmente solapadas. De este modo, se propone la creación de distintas matrices de partición Z_l utilizando los primeros l eigenvectores con $1 < l \leq min(n, m)$. Excluimos de este proceso el primer eigenvector z_1, que es constante y por tanto no aporta información sobre cómo particionar G. Esta opción es factible puesto que las matrices de expresión normalmente tienen un número de columnas o muestras m del orden de decenas, mucho menor que el número de filas o genes n.

En la Sección 3.6 se muestra como las particiones obtenidas de Z_l pueden ser muy diferentes de las obtenidas de Z_{l-1} o Z_{l+1}, lo que demuestra que la utilización de distintas matrices de partición extiende el espacio de soluciones posibles. Los resultados experimentales muestran, además, que cuando l aumenta y se acerca a su valor máximo de $min(n, m)$, la calidad de los biclusters obtenidos empeora, debido a que los últimos eigenvectores representan mayoritariamente el ruido de los datos de partida. La elección de un valor adecuado de l depende del *dataset* concreto que se esté analizando. En la Sección 3.6 mostramos la evolución de la calidad de los biclusters obtenidos a partir de distintos valores de l para los *datasets* estudiados.

3.5.3 Agrupamiento de filas y columnas independientemente.

La aplicación del algoritmo de clustering simultáneamente a filas y columnas de A (a todas las filas de la matriz de partición o espectro Z), no produce resultados satisfactorios cuando el número de filas n de A es mucho mayor que el número de columnas m de A. Éste es precisamente el caso de las matrices de expresión genética, donde se dispone de valores de expresión para, típicamente, miles de genes bajo decenas de condiciones experimentales. En estas condiciones, el algoritmo de clustering posibilístico tiende a asignar valores bajos de pertenencia a todas las columnas en todos los grupos, con lo que no aporta información sobre como agrupar las columnas en los distintos biclusters.

Otra limitación que presenta esta estrategia es que los grados de pertenencia de los genes (filas) a un bicluster posibilístico no son independientes de los grados de pertenencia de las muestras (columnas), y viceversa. Sin embargo, tratar los grados de pertenencia de forma independiente para genes y muestras sería deseable dada la naturaleza de los datos considerados.

Para solventar estas limitaciones se propone la aplicación del algoritmo de clustering de forma independiente sobre los genes (n primeras filas del espectro Z_l) y las muestras (m últimas filas de Z_l). Combinando cada cluster posibilístico de genes obtenido con cada cluster posibilístico de muestras, se obtienen biclusters posibilísticos, como muestra la figura 3.1.

3.5.4 Biclusters crisp a partir de biclusters posibilísticos.

El algoritmo PSB utiliza el Residuo Cuadrático Medio (MSR) [69] como medida de calidad para seleccionar los mejores biclusters obtenidos y comparar los resultados con otros métodos. De acuerdo con el modelo aditivo $a_{ij} = \lambda + \phi_i + \gamma_j$, el residuo total de un bicluster (MSR), definido por las filas I y las columnas J es la media de los residuos al cuadrado de todos los elementos del bicluster:

$$MSR(I,J) = \frac{1}{|I|\,|J|} \sum_{i \in I, j \in J} r(a_{ij})^2 \tag{3.11}$$

El cálculo del MSR requiere transformar los biclusters posibilísticos en biclusters crisp. Para ello seleccionamos dos α-cortes: α_1 para los genes y α_2 para las muestras, tales que si incluimos en el bicluster crisp aquellos genes con valores de pertenencia

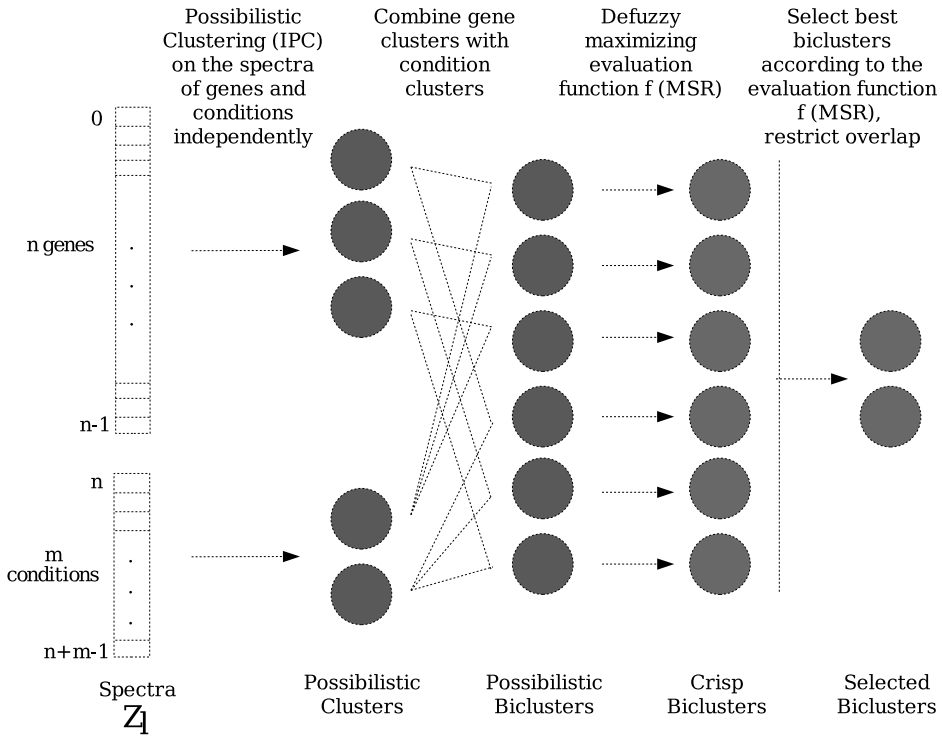


Figura 3.2: *Representación gráfica de la parte principal del algoritmo PSB (pasos 2(a), 2(b) y 2(c) del pseudocódigo de PSB 3.5.7).*

mayores que α_1, y aquellas muestras con valores de pertenencia mayores que α_2, optimizamos una función objetivo f.

Definimos f en función del MSR y del volumen ($Num.filas \times Num.columnas$ o $size_f \times size_c$) del bicluster. Una alternativa sería definir:

$$f(MSR, size_f, size_c) = \frac{size_f \cdot size_c}{MSR}$$

Sin embargo, esta función de optimización produce biclusters con un residuo demasiado alto, que no resultan interesantes desde un punto de vista biológico a pesar de su gran tamaño.

Experimentalmente, hemos comprobado que la siguiente función produce biclusters de buena calidad y tamaño:

$$f = g \times h_f \times h_c$$

donde:

- g es una función exponencial decreciente del MSR, que toma valor 1 cuando MSR es 0 y tiende a 0 cuando el MSR crece: $lim_{x \to +\infty} g(x) = 0$;

- h_f y h_c son funciones exponenciales que reciben el número de filas y columnas, respectivamente, como entrada, y toman valor 0 cuando el número de filas (columnas) es 0 y valor 1 cuando el número de filas (columnas) crece: $lim_{x\to+\infty}h_f(x) = 1$.

3.5.5 Limitación de solapamiento entre biclusters.

Aunque el algoritmo IPC reduce el número de biclusters con muy alto grado de solapamiento, algunos biclusters altamente solapados pueden ser obtenidos. Para evitar este excesivo solapamiento, comprobamos si los biclusters que se obtienen tienen un grado de solapamiento mayor que un determinado porcentaje q con alguno de los biclusters ya obtenidos. Si dos biclusters se solapan en un porcentaje mayor a q, seleccionamos únicamente el mejor de los dos para permanecer en el conjunto de biclusters que se devuelven al usuario como salida del algoritmo, descartando el otro bicluster. En otro caso, reemplazamos el peor bicluster del conjunto con el nuevo bicluster obtenido.

3.5.6 Inversión de la matriz de expresión.

El enfoque propuesto tiene como objetivo particionar el grafo que representa a la matriz de expresión de forma que se maximicen los pesos de los arcos dentro de cada subgrafo (es decir, minimizando el corte del grafo). Esta aproximación produce biclusters de genes sobre-expresados, ya que maximizar los pesos de los arcos dentro de los subgrafos equivale a maximizar los valores de expresión de los genes de los biclusters. Sin embargo, cuando se trabaja con matrices de expresión genética, también resulta interesante obtener biclusters de genes sub-expresados, es decir, con bajos valores de expresión para un subconjunto de muestras. Para obtener este tipo de biclusters, se invierte linealmente la matriz $A_{n\times m}$, obteniendo $A'_{m\times n}$. A continuación, se aplica el algoritmo PSB sobre la nueva matriz $A'_{m\times n}$. Los biclusters obtenidos de la aplicación de PSB sobre A' identifican genes sub-expresados en la matriz de expresión original A.

3.5.7 Pseudocódigo del algoritmo PSB.

1. Aplicar el método SVD sobre $A_n = D_1^{-\frac{1}{2}}AD_2^{-\frac{1}{2}}$ para resolver el eigenproblema $Lz = \lambda Dz$. Obtener $min(n, m)$ eigenvectores solución z.
2. Para $l = 1..s$ construir la matriz de partición Z_l cuyas columnas son los eigenvectores $z_2..z_{l+1}$. El algoritmo puede ser detenido cuando los biclusters obtenidos a partir de Z_s presentan peor calidad que los obtenidos de Z_{s-1}.

a) Aplicar clustering posibilístico a las primeras n filas de Z_l, para obtener una partición posibilística de los nodos del grafo que representan genes (filas de A).

b) Aplicar clustering posibilístico a las últimas m filas de Z_l, para obtener una partición posibilística de los nodos del grafo que representan muestras (columnas de A).

c) Combinar cada cluster de genes obtenido C_r y cada cluster de muestras C_c, para obtener un bicluster posibilístico B_{pos}.

   i. Calcular el bicluster crisp B_{crisp} a partir de B_{pos}: Buscar el α_1-corte para C_r y el α_2-corte para C_c que maximicen la función de calidad f para B_{crisp}.

   ii. Conservar B_{crisp} si:

      A. Si B_{crisp} es mejor que el peor bicluster conservado hasta el momento y no se solapa en más de q con ningún otro bicluster conservado. En este caso, B_{crisp} reemplaza al peor bicluster.

      B. Si B_{crisp} se solapa en más de q con otro bicluster pero B_{crisp} es de mejor calidad. En este caso, reemplazar el bicluster solapado por B_{crisp}.

3. Aplicar el mismo algoritmo sobre la matriz invertida: aplicar SVD sobre la matriz $A'_{m\times n}$, evaluando la calidad de los biclusters obtenidos sobre $A_{n\times m}$.

3.6 Experimentos y análisis de resultados.

En esta sección presentamos los resultados obtenidos por el algoritmo propuesto sobre datos de expresión genética artificiales y reales. Además, se presenta una evaluación comparativa de los resultados obtenidos por PSB y los obtenidos mediante otros algoritmos de biclustering que utilizan el MSR como criterio de optimización y que también permiten identificar biclusters arbitrariamente posicionados en la matriz de expresión (biclusters solapados). Estos algoritmos son los desarrollados por Cheng y Church [69] y el algoritmo FLOC de Yang *et al.* [334]. Centraremos la comparativa en el MSR y tamaño de los biclusters obtenidos. Además, hemos evaluado la bondad de PSB en comparación con el algoritmo PLAID [195], de uso muy extendido en la comunidad, que también adopta la definición de biclusters basada en modelos aditivos, aunque no utiliza el Residuo Cuadrático Medio como medida de calidad. En este caso,

centraremos la evaluación comparativa en la significación biológica de los biclusters obtenidos.

La experimentación utiliza *datasets* sintéticos y reales. Los *datasets* sintéticos permiten evaluar la bondad de los distintos algoritmos de biclustering en un entorno controlado, en el que se conocen los patrones que existen en las matrices de expresión. Los *datasets* reales nos permiten evaluar el potencial de estos mismos algoritmos en casos reales y su capacidad para capturar patrones con significación biológica. Los datasets reales que se emplean en esta experimentación son los mismos que se emplearon en la Sección 2.8: el *dataset* de la levadura, con 2879 genes bajo 17 condiciones relacionadas con el ciclo celular de *Saccharomyces cerevisiae* [71]; y el *dataset* de linfomas, con 4026 genes bajo 96 muestras tisulares de distintos tipos de *H.Sapiens* [25].

Los biclusters obtenidos de la aplicación del algoritmo de biclustering de Cheng y Church [69] para el *dataset* de la levadura y el *dataset* de linfomas están disponibles en `http://cheng.ececs.uc.edu/biclustering`. En el caso del algoritmo FLOC [334], sólo se dispone de información del residuo y volumen promedio de los biclusters obtenidos sobre estos *datasets*. Para efectuar una comparativa amplia y detallada, se ha implementado el algoritmo FLOC y se ha ajustado para producir resultados similares a los obtenidos en [334]. Para evaluar el algoritmo de Cheng y Church en los datos sintéticos, hemos utilizado el software BicAt [42]. Finalmente, para evaluar el algoritmo PLAID, se ha utilizado la implementación disponible en `http://www-stat.stanford.edu/~owen/plaid/`.

3.6.0.1 Datos sintéticos

Dataset con un bicluster.

El primer caso de estudio consiste en introducir un bicluster de 200 genes por 4 muestras en un *dataset* de tamaño 1000 por 12. El bicluster está formado por valores de expresión altos (genes sobre-expresados) generados de acuerdo al modelo aditivo, de forma que el residuo del bicluster es 0. El resto de la matriz ha sido rellenada con ruido generado a partir de una distribución uniforme. Se han generado 10 matrices siguiendo este proceso y ejecutado los distintos algoritmos de biclustering sobre todas ellas. La tabla 3.1 muestra, para el mejor bicluster obtenido para cada método, el porcentaje de su volumen (en media) que se corresponde y no se corresponde, respectivamente, con el bicluster real. Estos valores se muestran en la tabla como "%

Algoritmo	%Volumen Compartido (Avg.)	%Volumen No Compartido(Avg.)
CC	44.51 %	67.06 %
FLOC	39.9 %	37.62 %
PLAID	13.75 %	86.27 %
PSB	100 %	2.85 %

Cuadro 3.1: *Comparativa de resultados de algoritmos de biclustering sobre datos sintéticos con un bicluster. "% Volumen Compartido" indica el porcentaje de volumen del bicluster encontrado que se corresponde con el bicluster real. "% Volumen No Compartido" indica el porcentaje de volumen del bicluster encontrado que no se corresponde con el bicluster real.*

Volumen Compartido" para referirnos al porcentaje de volumen del bicluster encontrado que es compartido por el bicluster real y "% Volumen No Compartido" para referirnos al porcentaje de volumen del bicluster encontrado que no se corresponde con el bicluster real. Podemos apreciar que en todos los casos, PSB obtuvo un bicluster cubriendo por completo (100 %) el volumen del bicluster real, con solo un 2,85 % de volumen adicional (en media). Los métodos de Cheng y Church, FLOC y PLAID identificaron principalmente patrones espúreos del fondo ruidoso.

Dataset con múltiples biclusters.

Para evaluar la capacidad de los distintos algoritmos de identificar múltiples biclusters, especialmente cuando éstos pueden estar solapados, presentamos este caso de estudio en el que introducimos tres biclusters constantes de volumen 200×4 posicionados arbitrariamente en una matriz de tamaño 1000×12. Se han generado 10 datasets siguiendo este procedimiento y ejecutado los algoritmos de biclustering sobre ellos. Para evaluar los resultados elegimos los tres biclusters proporcionados por cada método que mejor se corresponden con los tres biclusters reales introducidos en el *dataset*. La Tabla 3.2 muestra el porcentaje promedio de los biclusters resultantes que es compartido y no compartido por los biclusters reales. De nuevo, PSB presenta la mayor cobertura de los biclusters reales y el menor porcentaje de volumen desperdiciado en patrones espúreos.

3.6.0.2 Datos de expresión reales.

En esta sección se presenta una evaluación comparativa del PSB y los distintos algoritmos de biclustering comentados a lo largo de esta sección sobre los dos *datasets* reales que ya se emplearon en la Sección 2.8: los datos de expresión del ciclo celular

Algorithm	%Volumen Compartido (Avg.)	%Volumen No Compartido (Avg.)
CC	39.3 %	81.38 %
FLOC	48.04 %	32.4 %
PLAID	25.12 %	71.52 %
PSB	83.78 %	13.1 %

Cuadro 3.2: *Comparativa de resultados de algoritmos de biclustering sobre datos sintéticos con tres biclusters. "% Volumen Compartido" indica el porcentaje de volumen de los biclusters encontrados que se corresponde con los biclusters reales. "% Volumen No Compartido" indica el porcentaje de volumen de los biclusters encontrados que no se corresponde con los biclusters reales.*

de la *S. cerevisiae* obtenidos de [71] (al que nos referimos como *dataset de la levadura*) y los datos de expresión humanos de distintos tipos de linfomas obtenido de [25] (al que nos referimos como *dataset linfoma*). Como ya se comentó en la Sección 2.8, el dataset linfoma resulta especialmente adecuado para evaluar la validez de las técnicas de biclustering, ya que contienen un gran número de muestras y de gran heterogeneidad (nueve tipos distintos de linfomas y tejidos sanos). Los algoritmos de biclustering pueden ser utilizados para encontrar patrones de expresión que afecten únicamente a uno o varios subtipos de muestras.

Centraremos las comparaciones entre los distintos algoritmos de biclustering en el residuo cuadrático medio (MSR) y el volumen de los 100 mejores biclusters obtenidos por cada algoritmo. Además, como ya se propuso en la Sección 2.8, utilizamos las anotaciones de *Gene Ontology* para determinar si los genes de un mismo grupo tienen una misma función biológica. En particular, para comparar la significación biológica de los resultados obtenidos por los distintos métodos, utilizaremos *correspondence plots* [298]. Estos diagramas representan la distribución de p-valores de los biclusters obtenidos: para cada valor de p, el porcentaje de biclusters obtenidos por el método cuyo p-valor es, al menos, igual a p.

Dataset de la levadura

El dataset de la levadura publicado en [69] contiene el nivel de expresión de 2879 genes de la levadura bajo 17 condiciones experimentales relacionadas con el ciclo celular (cubriendo aproximadamente dos ciclos celulares completos). Los datos han sido seleccionados y preprocesados de acuerdo con [69].

Elección del número de eigenvectores. La Figura 3.3 muestra el promedio del número de biclusters y el promedio para la función de calidad f de los biclusters obte-

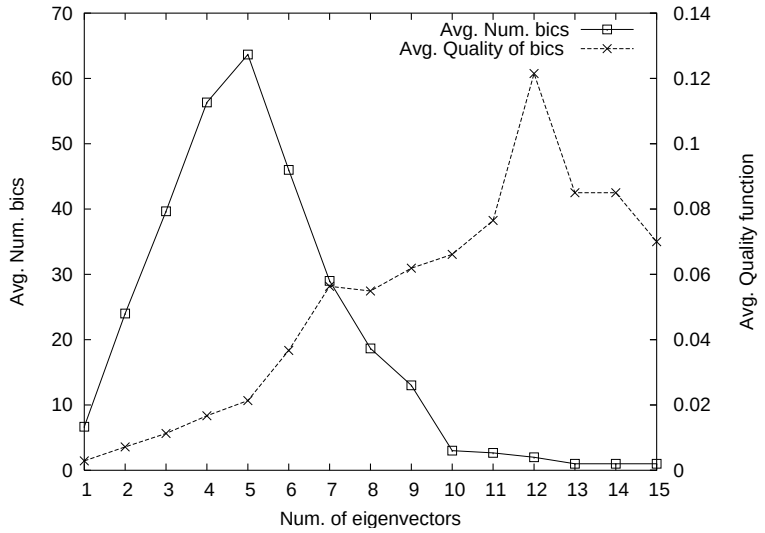


Figura 3.3: *Evolución del número y calidad de los biclusters obtenidos en el dataset de la levadura utilizando de 1 a 15 eigenvectores.*

Cuadro 3.3: *Resultados de PSB y comparativa con otros métodos para el dataset de la levadura.*

Algorithm	Residue Avg.	Rows Avg.	Cols. Avg.
Random	1009.84	300	8
CC	204.293	167	12
FLOC	187.543	195	12.8
PLAID	-	462.73	7.40
PSB	169.03	274.42	7.42

nidos aplicando PSB a los datos de la levadura, utilizando las matrices de partición de Z_1 a Z_{15}. La evolución de ambas representaciones indica que las matrices de partición con más de 12 eigenvectores producen un descenso en la calidad de los biclusters obtenidos en este *dataset*, por lo que establecemos $s = 12$ para este *dataset*.

Resultados experimentales. La Tabla 3.3 muestra el resido promedio y número promedio de filas y columnas para los 100 mejores biclusters obtenidos por los algoritmos PSB, FLOC, Cheng y Church (CC) y PLAID. También mostramos los valores obtenidos por 100 biclusters aleatorios de volumen similar a los obtenidos por el PSB (300×8). Para los resultados del algoritmo PLAID, recordamos que no mostramos el valor del Residuo Cuadrático Medio puesto que PLAID no utiliza este criterio de optimización.

PSB supera los resultados de los algoritmos Cheng&Church y FLOC en términos del residuo y número promedio de filas, aunque el número promedio de columnas es

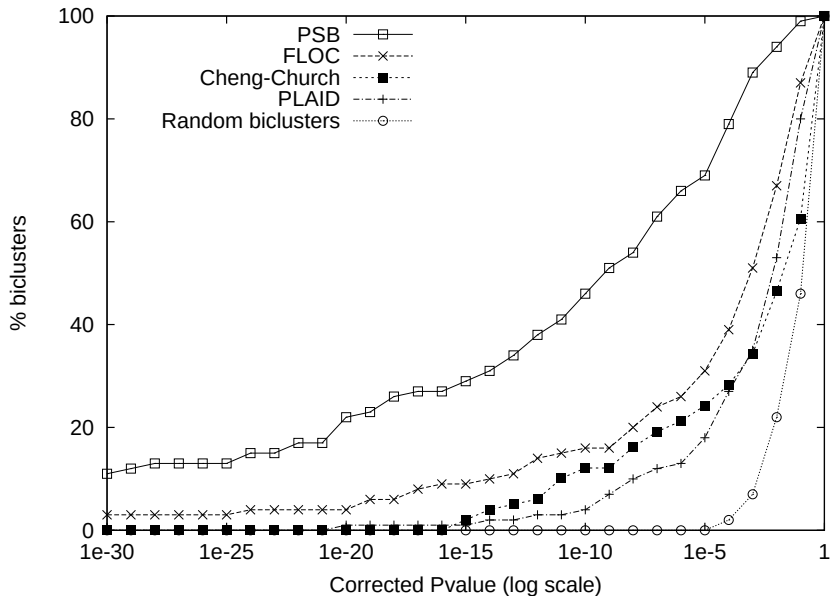


Figura 3.4: *Correspondence plots para biclusters generados con PSB, FLOC, Cheng&Church, PLAID y biclusters aleatorios en los datos de la levadura.*

menor. Sin embargo, la función f para seleccionar biclusters puede adaptarse para obtener patrones que involucren un mayor número de columnas, en detrimento de la calidad de los patrones.

Los *correspondence plots* para los biclusters obtenidos con cada método se representan en la Figura 3.4. Podemos comprobar que sea cual sea el valor umbral de significación que consideremos, PSB obtiene el mayor porcentaje de biclusters con un p-valor corregido menor que dicho umbral. Por ejemplo, si establecemos el umbral de significación en 10^{-5}, PSB obtiene un promedio de 69 biclusters (del total de 100 identificados) que tienen alguna función biológica asociada significativamente (de entre las disponibles en la ontología GO) con un p-valor corregido inferior a 10^{-5}. Los demás algoritmos encuentran 31 biclusters significativamente enriquecidos (FLOC), 24 (Cheng&Church) y 18 (PLAID) para estos mismos valores de significación. Este experimento nos permite concluir que el algoritmo PSB agrupa conjuntamente más genes con una función biológica más relacionada, por lo que los biclusters obtenidos por este algoritmo serán más informativos y fiables desde un punto de vista biológico.

Otra prueba de validación de los resultados de PSB y de la metodología utilizada se muestra en la Figura 3.5. Esta figura muestra los *correspondence plots* asociados a los resultados obtenidos por el algoritmo PSB reemplazando, al azar, una fracción aleatoria de los genes de cada bicluster por otros genes escogidos al azar de la matriz

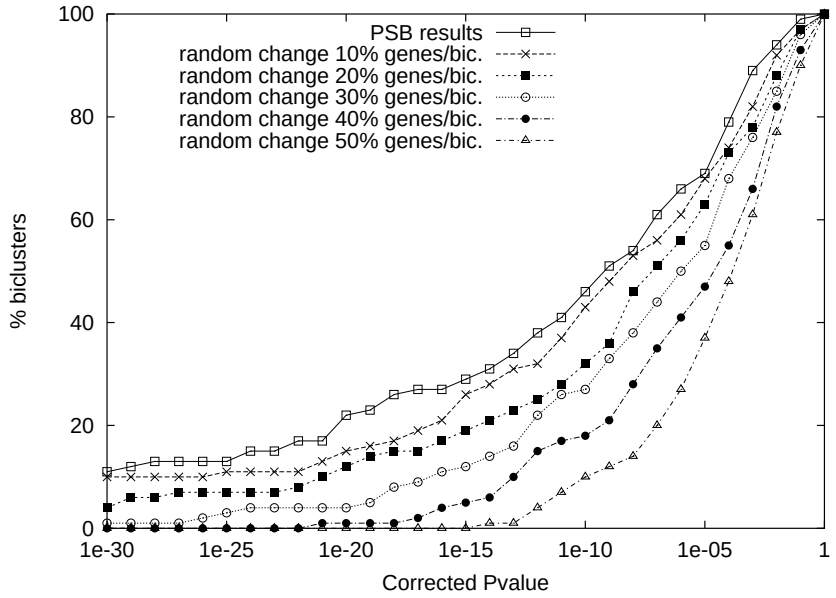


Figura 3.5: *Correspondence plots para biclusters obtenidos reemplazando al azar el 10,20,30,40 y 50 % de los genes de cada bicluster obtenido por PSB en el dataset de la levadura.*

Cuadro 3.4: *Procesos biológicos de GO más significativos asociados a algunos de los 100 biclusters obtenidos por PSB para el dataset de la levadura.*

Bicluster	GO term	Corrected p-value
48	macromolecule biosynthesis	1.88e-44
50	cell organization and biogenesis	1.06e-11
51	ribosome biogenesis	9.72e-19
61	protein biosynthesis	7.18e-32
68	primary metabolism	2.19e-21
94	nucleobase, nucleoside, nucleotide and nucleic acid metabolism	5.16e-08

de expresión. Los *correspondence plots* muestran que la significación biológica de los biclusters decrece aunque el cambio sólo afecte a un pequeño porcentaje de los genes de cada bicluster.

La metodología utilizada para validar los resultados demuestran la potencia y validez de los resultados obtenidos por el algoritmo PSB. A modo ilustrativo, la Tabla 3.4 muestra los términos GO más significativos para algunos de los biclusters obtenidos por PSB.

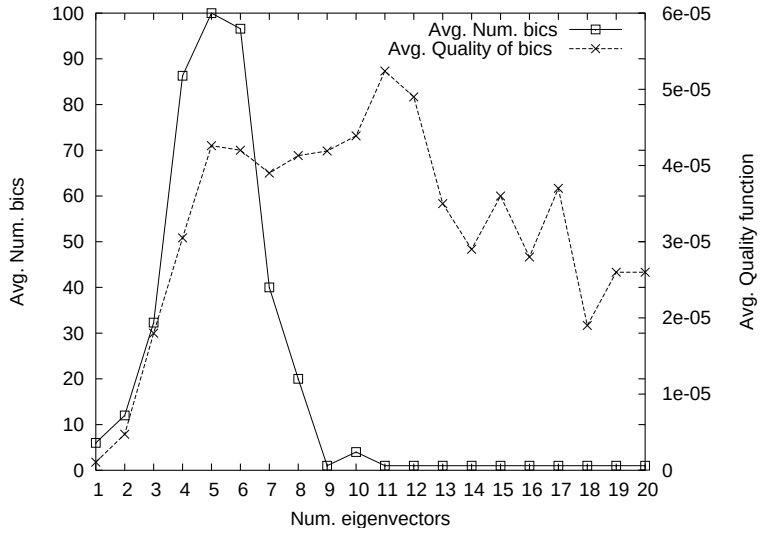


Figura 3.6: *Evolución del número y calidad de los biclusters obtenidos en el dataset de linfomas utilizando de 1 a 20 eigenvectores.*

Dataset de Linfomas.

El dataset de Linfomas contiene los niveles de expresión de 4026 genes para 96 muestras de tejido humano, que se clasifican en 9 tipos distintos de linfomas y tejidos sanos [25]. Este dataset ha sido descargado de la web de material suplementario de [25] y ha sido transformado para eliminar valores de expresión negativos. Esta transformación consiste en trasladar los valores originales, que están expresados como ratio logarítmico y tienen un rango [-7.5,6.5], hacia valores positivos utilizando:

$$X' = (log_2((2^X) * 180)) * 40$$

donde X son los valores de expresión originales proporcionados en ratio log_2.
De este modo, la matriz resultante se ajusta aproximadamente al rango [0,600]. Los valores perdidos fueron reemplazados por el valor promedio inferido de los 15 vecinos más cercanos utilizando la distancia Euclídea.

Elección del número de eigenvectores. La Figura 3.6 muestra el promedio del número de biclusters y la calidad de los mismos obtenidos aplicando el PSB a los datos de linfoma humano utilizando las matrices de partición Z_1 a Z_{20}. De nuevo, 12 parece un buen valor para s ya que las matrices de partición Z_1 a Z_{12} producen los biclusters de mejor calidad.

Algorithm	Residue Avg.	Rows Avg.	Cols. Avg.
Random	930.33	960	50
CC	850.04	269.2	24.5
FLOC	379.62	892.53	42.12
PLAID	-	1003.63	41.7
PSB	361.4	965.1	49.5

Cuadro 3.5: *Resultados de PSB y comparativa con otros métodos en el dataset de linfomas.*

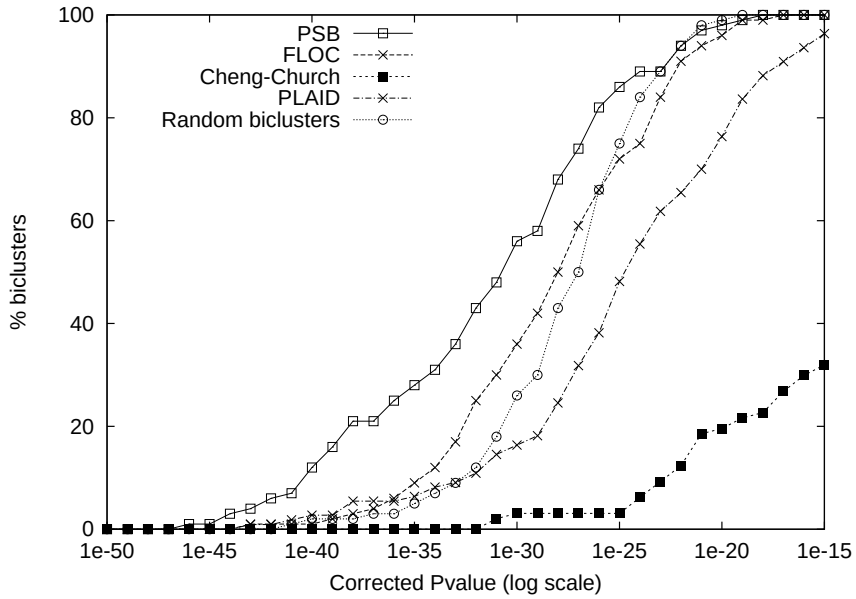


Figura 3.7: *Correspondence plots para biclusters obtenidos con PSB, FLOC, Cheng&Church, PLAID y biclusters aleatorios sobre dataset de linfomas.*

Resultados Experimentales. Como muestra la Tabla 3.5, PSB obtiene menor MSR para los biclusters que los algoritmos FLOC y Cheng & Church. El resido promedio de biclusters aleatorios de tamaño similar a los obtenidos por PSB (960×50) también es mostrado en la tabla.

Los *Correspondence plots* para los biclusters obtenidos con cada uno de los métodos se muestran en la Figura 3.7. La distribución de p-valores asociada a los resultados de los distintos algoritmos muestra que los genes de los biclusters de PSB están más relacionados biológicamente entre sí que los genes de los biclusters obtenidos por FLOC, Cheng & Church, PLAID y biclusters aleatorios.

En este caso, sin embargo, las diferencias entre los biclusters aleatorios y los otros métodos no son tan significativas como lo fueron con el *dataset* de la levadura. Esto era previsible, puesto que se dispone de menos información para los genes de

H.Sapiens que para los de *S. Cerevisiae*. Por ejemplo, de los 2879 genes totales incluidos en el dataset de la levadura, 2601 tienen una función biológica conocida y anotada en la ontología Gene Ontology. Sin embargo, para los genes del dataset de linfomas humano, solo 2228 de los 4026 genes tienen una función biológica conocida.

Sin embargo, incluso en este tipo de datasets donde el conocimiento biológico conocido es menor, PSB proporciona los biclusters más enriquecidos funcionalmente, dando muestra de su capacidad para descubrir patrones relevantes en los datos de expresión.

Por último, para los datos de linfomas humanos, podemos considerar también la interpretación biológica de las muestras de los biclusters ya que las muestras de la matriz se clasifican en distintos tipos de linfomas y tejidos sanos [25]. Para determinar si los biclusters obtenidos por PSB y los demás métodos de biclustering respetan esta clasificación de muestras y obtienen biclusters en los que un tipo de muestras está significativamente representado, hemos calculado la significación estadística de los tipos de muestras en cada bicluster. Los *correspondence plots* de la Figura 3.8 muestran esta representación significativa de tipos de muestras en biclusters para los métodos PSB, FLOC, Cheng & Church, PLAID y biclusters aleatorios. Los resultados muestran que los biclusters obtenidos por PSB se acogen mejor a esta clasificación conocida de las muestras. Por ejemplo, un bicluster identificado por PSB permite caracterizar un tipo de linfoma concreto: *Diffuse Large B-Cell Lymphoma* y diferenciarlo de otros tejidos. Otro biclusters obtenidos confirman similitudes previamente corroboradas y publicadas en [25] como aquellas entre los tipos *Activated blood B*, *Resting blood B* y algunos tejidos *DLBCL*; o entre *Chronic Lymphocytic Leukaemia*, *Follicular Lymphoma* y *Resting blood B*. Los biclusters mencionados y todos los resultados obtenidos utilizando PSB están disponibles en `http://decsai.ugr.es/~ccano/psb/`.

3.7 Conclusiones.

En este capítulo se ha presentado un método biclustering denominado Biclustering Espectral Posibilístico (PSB) para el análisis de matrices de expresión genética basado en tecnología difusa y técnicas espectrales de clustering. El PSB utiliza el residuo cuadrático medio (MSR) como criterio de optimización, y permite identificar biclusters potencialmente solapados en la matriz de expresión. Los resultados experimentales obtenidos en *datasets* sintéticos y reales, han demostrado que el PSB mejora los resultados de otros algoritmos de similares características, como los algoritmos de Cheng & Church y FLOC, en términos de MSR de los biclusters obtenidos.

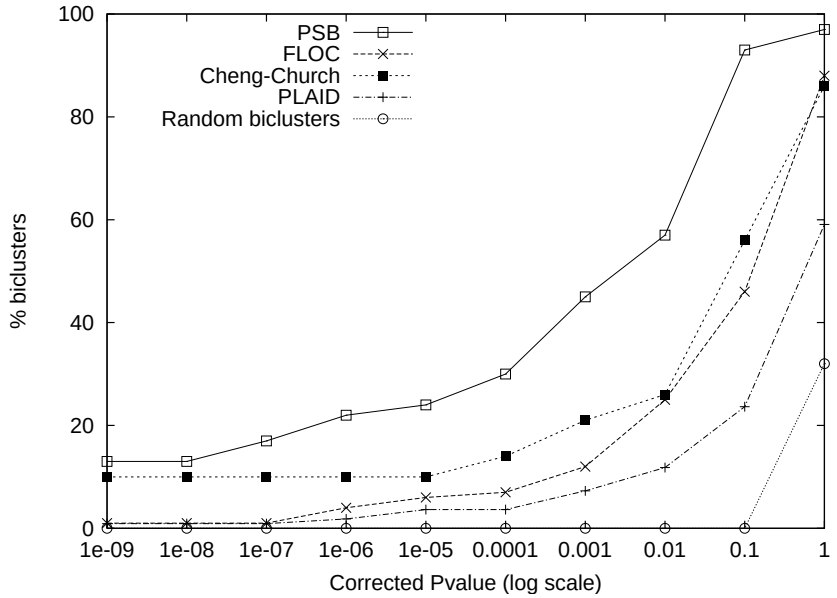


Figura 3.8: *Correspondence plots para la significación de tipos de muestras en los biclusters obtenidos por PSB, FLOC, Cheng&Church, PLAID y biclusters aleatorios para el dataset de linfomas. En cada caso se selecciona el tipo de condición con el menor p-valor para cada bicluster.*

Además, el algoritmo propuesto identifica patrones con mayor significación biológica que los obtenidos utilizando los algoritmos de Cheng & Church, FLOC o PLAID. Hemos comprobado que, en efecto, PSB agrupa conjuntamente genes más íntimamente ligados respecto a su función biológica, de acuerdo con la información proporcionada por la ontología *biological process* de *Gene Ontology*, tanto para el *dataset de la levadura* como para el *dataset de linfomas*. Del mismo modo, los resultados de PSB se ajustan mejor a la clasificación de muestras proporcionada por Alizadeh *et al*. [25] para el *dataset de linfomas*.

La validación biológica que hemos presentado en este capítulo nos muestra el potencial del biclustering. Los procesos biológicos estadísticamente significativos asociados a los genes de un bicluster pueden proporcionar mucha información acerca de las propiedades que caracterizan las muestras agrupadas en ese bicluster.

El trabajo futuro en esta sección irá encaminado a la integración de información de distintas fuentes de datos biológicas, como las matrices de expresión, las bio-ontologías, y la información de la literatura, para obtener nuevos descubrimientos relevantes desde un punto de vista biológico.

Parte III

Anotación manual y sistemas automáticos para la extracción de conocimiento de textos biomédicos

Capítulo

4

Sistemas de anotación colaborativa de textos biomédicos

4.1 Motivación y objetivos

La literatura biomédica crece a ritmo exponencial [149], y en la actualidad, MEDLINE contiene más de 19 millones de publicaciones, de las cuales más de 3.5 millones fueron publicadas en los últimos 5 años [183]. El conjunto de la literatura biomédica, habitualmente denominado *biblioma* (*bibliome*), representa pues un recurso de enorme potencial y riqueza para comprender la base genética de las enfermedades, ya que contiene información fiable y de calidad resultado de años de esfuerzo e investigación de la comunidad científica. A modo de ejemplo, la literatura biomédica constituye la principal fuente de conocimiento sobre los genes y proteínas, su papel en distintas enfermedades, mecanismos de interacción con otros elementos biológicos, información estructural, etc.

Recientes estudios sobre redes de interacción genética aplicados a la ataxia [201, 148] y la enfermedad de Huntington [121] han demostrado que la integración de conocimiento biológico de distintas fuentes de datos puede conducir al descubrimiento de nuevas hipótesis y hechos biológicos de relevancia para el entendimiento de los mecanismos reguladores de estas enfermedades. Resulta particularmente atractivo en estos estudios, relacionar conceptos utilizando el extenso conocimiento almacenado en la literatura especializada. Por ejemplo, cuando se genera una lista de genes como resultado de un estudio clínico sobre una enfermedad, es necesario situarlos en un contexto más amplio de mecanismos celulares e interacciones moleculares para un mejor entendimiento de los procesos biológicos subyacentes.

Existen numerosos esfuerzos públicos y privados orientados a la extracción de conocimiento de la literatura biomédica y la representación del mismo en forma de entidades biológicas y relaciones entre las mismas. Como fruto de estos esfuerzos, una parte del conocimiento disponible en la literatura científica se encuentra almacenado de forma estructurada en bases de datos (como Uniprot, RefSeq, EntrezGene, Model Organism databases, KEGG, GeneCards) y ontologías especializadas (como Gene Ontology). Estos recursos permiten un acceso eficiente a los datos y facilitan su manejo y utilización para una amplia variedad de estudios y análisis. El contenido de la mayor parte de estos recursos es extraído de la literatura por medio de anotadores humanos, por lo que, aunque de extraordinario valor, estos recursos sólo recogen una pequeña fracción de la información publicada en la literatura, y requieren de extensos esfuerzos para ser actualizados y ampliados. Los recursos *poblados* por medios automáticos habitualmente presentan un mayor tamaño y cobertura de la realidad biológica que modelan, pero también muestran elevadas tasas de imprecisión y errores [261]. Además, el desarrollo y mejora de plataformas de text-mining requiere, a su vez, de corpora anotado de forma manual, para el entrenamiento y test de los mecanismos de aprendizaje automático que soportan estas plataformas.

Esta situación hace que frecuentemente el investigador clínico deba buscar manualmente la evidencia biológica que le resulta de interés y utilizar métodos *ad-hoc* para representar este conocimiento.

En este capítulo se presenta una herramienta que permite combinar de forma efectiva la anotación manual de textos y las técnicas de text-mining, para una extracción de conocimiento más eficiente de los textos biomédicos, permitiendo simultáneamente el desarrollo y mejora de técnicas avanzadas de text-mining.

En los últimos tiempos, con el auge de Internet y las comunidades virtuales, los proyectos colaborativos a gran escala están recibiendo una mayor difusión, aceptación y apoyo. Este tipo de esfuerzos muestran un enorme potencial, ya que permiten a millones de usuarios de todo el mundo colaborar en el desarrollo de un proyecto determinado, en este caso, la anotación de textos biomédicos. La plataforma Mechanical Turk de Amazon Web Services (`http://aws.amazon.com/`) es un ejemplo de este tipo de servicios basados en esfuerzos distribuidos y comunitarios que ha sido acogido con una gran expectación por la comunidad. La anotación colaborativa o comunitaria también ha sido adoptada recientemente por la comunidad biomédica. Por ejemplo, WikiProteins [229] o WikiGene [211] proporcionan entornos apropiados para la anotación comunitaria de genes y proteínas, entre otras entidades biológicas de interés.

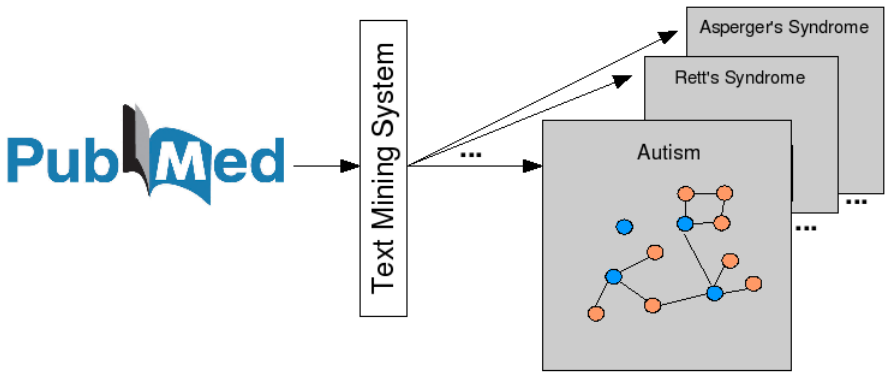


Figura 4.1: *Extracción de redes de interacción entre entidades biomédicas (genes, proteínas, enfermedades, fármacos, etc.) por medio del análisis de la literatura.*

Sin embargo, mientras que estos esfuerzos proporcionan a la comunidad con un medio para acceder y compartir gran cantidad de información indexada por entidades biológicas de interés, no están diseñados para servir en la creación de un corpus de texto, que explícitamente constate estructuras y entidades en el texto que sean útiles para el entrenamiento y validación de herramientas de text-mining. De este modo, crece la necesidad de un entorno colaborativo, en el que la anotación manual y automática sean combinadas de forma efectiva y eficiente para una mejor extracción de conocimiento de la literatura.

En este capítulo proponemos una herramienta que combina la anotación colaborativa de textos biomédicos y sistemas automáticos de extracción de conocimiento. Este trabajo constituye un primer paso hacia la construcción de un sistema en el que las redes de interacción entre entidades biológicas extraídas manualmente de la literatura (ver Figura 4.1), son enriquecidas con anotaciones sintácticas y semánticas del texto que evidencia y soporta las relaciones representadas en dichas redes. Estas anotaciones sirven para desarrollar y mejorar los métodos de text-mining que, a su vez, son utilizados para identificar relaciones candidatas y facilitar así el proceso de anotación manual.

El sistema propuesto ofrece un entorno intuitivo para la anotación distribuida de extractos de textos biomédicos. Aunque inicialmente el sistema fue concebido para la anotación manual de relaciones gen-gen y gen-enfermedad, éste fue pronto extendido para la anotación de cualquier tipo de entidad (genes, proteínas, enfermedades, fármacos, SNPs, términos de bio-ontologías, etc.) y cualquier tipo de relación entre las entidades de interés, aportando además, la posibilidad de integrar anotaciones

efectuadas por sistemas automáticos de text-mining al *pipeline* de anotación.

Este sistema, denominado BioNotate (`http://bionotate.sourceforge.net`), permite a distintos grupos de investigación llevar a cabo esfuerzos de anotación de corpora que se ajusten a sus necesidades particulares, contribuyendo simultáneamente a un esfuerzo de anotación a gran escala. En particular, existen dos niveles distintos de colaboración para los usuarios de BioNotate. Por una parte, distintos anotadores pueden colaborar en la anotación de un corpus alojado en sus propios servidores. Por otra parte, diversos corpora pueden ser anotados en distintos servidores, por distintos usuarios y grupos, y los corpora resultantes integrados en un único recurso común. De este modo, BioNotate proporciona una herramienta de anotación para sacar partido del enorme potencial colaborativo de la comunidad biomédica en internet y crear un corpus de gran tamaño para asistir en el desarrollo y mejora de herramientas de text-mining.

Además, dado que el conocimiento disponible en la literatura científica se encuentra almacenado en un amplio conjunto de bases de datos y ontologías, y que la necesidad de combinar datos de distintas fuentes es pues manifiesta, la difusión de anotaciones utilizando representaciones, protocolos y tecnologías estándares resulta fundamental para que la integración de este conocimiento con el de otros recursos comunitarios sea factible. Como se introdujo en 1.4.2, la Web Semántica ofrece un marco social y técnico para la integración y distribución de conocimiento biomédico a través de la Web, proponiendo lenguajes de representación, protocolos web y tecnologías estándares para la integración de información. BioNotate incorpora estas tecnologías para permitir un mejor acceso y recuperación de la información anotada, y una mayor conectividad con otros recursos del campo.

El capítulo se estructura como sigue. La sección 4.2 revisa los corpora disponibles relacionados con la anotación de entidades (genes, proteínas, enfermedades) y relaciones entre las mismas en la literatura biomédica, justificando la necesidad de un nuevo paradigma colaborativo para la anotación de corpora de mayor tamaño, y siguiendo esquemas de anotación adecuados. En la sección 4.3 se revisan herramientas existentes para la anotación de textos biomédicos, además de otras herramientas y filosofías de anotación aplicadas a otros campos que han servido de inspiración para la creación de BioNotate. La sección 4.4 describe el sistema de anotación BioNotate tal y como fue originariamente concebido, es decir, como sistema colaborativo para la anotación de relaciones gen-gen y gen-enfermedad en la literatura biomédica. En 4.5 se presenta un caso de estudio para la anotación de un corpora de relaciones

gen-gen para genes relacionados con el autismo, desarrollado en el marco del proyecto Autworks (http://autworks.hms.harvard.edu) en colaboración con el Centro de Informática BioMédica de la Universidad de Harvard, EEUU. La sección 4.6 presenta mejoras añadidas al sistema BioNotate original, así como otras aplicaciones desarrolladas para dominios específicos que utilizan BioNotate como sistema de anotación. Entre las mejoras y extensiones de BioNotate, caben destacar por su relevancia, la extensión que permite implementar cualquier esquema de anotación en BioNotate (descrita en 4.6.1), y la que integra anotaciones automáticas en BioNotate para permitir que los anotadores humanos corrijan las anotaciones efectuadas por sistemas de text-mining (4.6.3). Esta última extensión de BioNotate permite, además, difundir las anotaciones utilizando representaciones y protocolos de la Web Semántica. En 4.6.2 se describe AutismNotate, una aplicación desarrollada en colaboración con la Universidad de Harvard y la empresa Alias-I Inc., que utiliza BioNotate para la anotación de textos biomédicos relacionados con el Autismo. Finalmente, la sección 2.9 presenta las conclusiones y líneas de trabajo futuro relacionadas con el trabajo presentado en este capítulo.

4.2 Corpora en biomedicina

Para diseñar y entrenar un sistema de aprendizaje automático para identificar entidades de interés y relaciones entre las mismas en la literatura biomédica, es necesario contar con un corpus anotado con la información siguiente: ocurrencias de las entidades de interés, dependencias sintácticas o palabras clave de la interacción, las entidades que interaccionan y el tipo de interacción. En esta sección proponemos una revisión detenida de corpora públicos anotados manualmente con entidades biomédicas, sintaxis y relaciones entre entidades, prestando especial atención a los corpora anotados con genes/proteínas y relaciones proteína-proteína. Los corpora disponibles son los siguientes:

- Corpora con anotaciones de entidades biomédicas:
  - GENIA. `http://www-tsujii.is.s.u-tokyo.ac.jp/~genia/index.html`
  - GENIA-JNLPBA. `http://www-tsujii.is.s.u-tokyo.ac.jp/GENIA/ERtask/report.html`
  - GENETAG-05 (MedTag). `ftp://ftp.ncbi.nlm.nih.gov/pub/tanabe/GENETAG.tar.gz`
  - ABGene (MedTag). `ftp://ftp.ncbi.nlm.nih.gov/pub/tanabe/AbGene/`

  - Yapex. http://www.sics.se/humle/projects/prothalt/

- Corpora con anotaciones de relaciones proteína-proteína:
  - BioText. http://biotext.berkeley.edu/data.html
  - Wisconsin. http://www.biostat.wisc.edu/~craven/ie/
  - PICorpus. http://bionlp.sourceforge.net/PICorpus/index.shtml
  - Fetch Prot Corpus. http://www.sics.se/humle/projects/fetchprot/Corpus/Release20051011/
  - HIV-1 Human Protein Interaction. http://www.ncbi.nlm.nih.gov/RefSeq/HIVInteractions/index.html
  - BioCreAtIvE I - PPI. http://www2.informatik.hu-berlin.de/~hakenber/corpora/
  - SPIES Corpus. http://spies.cs.tsinghua.edu.cn
  - BioIE. http://bioie.biopathway.org/
  - BioContrasts. http://biocontrasts.i2r.a-star.edu.sg/BioContrasts-testcorpus.html
  - AIMED. ftp://ftp.cs.utexas.edu/pub/mooney/bio-data/
  - GENIA Events. http://www-tsujii.is.s.u-tokyo.ac.jp/GENIA/home/wiki.cgi?page=Event+Annotation
- Corpora con anotaciones de dependencias sintácticas:
  - PennBioIE. http://bioie.ldc.upenn.edu/
  - GENIA Treebank. http://www-tsujii.is.s.u-tokyo.ac.jp/~genia/index.html
  - Brown GENIA Treebank. http://bllip.cs.brown.edu/resources.shtml
  - DepGENIA. http://www.ifi.unizh.ch/cl/kalju/download/depgenia/
- Corpora con anotaciones de relaciones proteína-proteína y dependencias sintácticas:
  - BioInfer [255]. http://www.it.utu.fi/BioInfer
  - LLL05 (Genic Interaction Extraction Challenge LLL Workshop 05). http://genome.jouy.inra.fr/texte/LLLchallenge/

La tabla 4.1 muestra un resumen de las características más importantes de cada uno de ellos.

Type	Corpus Name	Object of the annotation	Length	+/−	Format
Named Entities	GENIA [173]	NE (concepts in GENIA ontology)	2000 abstracts		XML
	GENIA-JNLPBA [175]	NE(Proteins, DNAs, RNAs, cell lines, cell types)	404 abstracts		XML
	GENETAG-05 (MedTag) [297]	NE (gene/proteins)	15000sent		Own Stand-off
	ABGene (MedTag) [287]	NE (gene/proteins)	4265sent		Own
	Yapex [15]	NE (proteins)	200 abstracts		XML
Interactions	BioText [264]	PPI/Disease-Treatment	2143 interactions	+	Own
	Wisconsin [78]	PPI/Prot-Cell loc./Gene-Disease	52000/7900/13412sent	+/−	Own Stand-off
	PICorpus [161]	PPI	10271 sent	+	XML/WordFreak
	Fetch Prot Corpus [5]	PPI	190 full texts	+/−	Stand-off XML
	HIV-1 Human PI [9]	PPI	2224 interacting proteins	+	Own
	BioCreAtIvE I - PPI [141, 3]	PPI / NE	255 int/1000 sent	+/−	Stand-off XML
	SPIES Corpus [146]	PPI / NE	963 sent	+	Own
	BioIE [171]	PPI / NE	250 sent	+	HTML
	BioContrasts [172]	Prot-Prot contrasts	100 abstracts	+	HTML
	AIMED [1]	PPI / NE	225 abstracts	+/−	XML
	GENIA events [174]	PPI / NE	1000 abstracts	+/−	XML
	BioNotate [61]	PPI/Gene-Disease	-	+/−	Stand-off XML
Syntax	PennBioIE [13]	NE / Syntax (constituents)	642 abstracts		XML/WordFreak
	GENIA Treebank [302]	Syntax (constituents)	500 abstracts		XML/PTB
	Brown GENIA [196]	Syntax (constituents)	21 abstracts / 215 sent		PTB
	DepGENIA [4]	Syntax (dependencies)	All GENIA corpus		XML
Interactions & Syntax	BioInfer [255]	NE / PPI / Syntax (dependencies)	1100 sent / 2662 rel	+/−	XML
	LLL 05 [7]	NE / PPI / Syntax (dependencies)	80 sent	+	Own Stand-off

Cuadro 4.1: *Resumen de las características de los distintos corpora con anotaciones de entidades, sintácticas y relaciones proteína-proteína. Type: tipo de anotaciones (PPI para Interacción Proteína-Proteína, SYNTAX para dependencias sintácticas y PPI & SYNTAX para ambas). Object of the annotation: qué se anota en el corpus (PPI, Syntax: estructura sintáctica, NE: Named Entities o Entidades Biomédicas). Length: tamaño del corpus (en número de títulos, abstracts, frases, interacciones o textos completos). +/−: indica si se anotan los ejemplos en los que existe relación (positivos o +), aquellos en los que no existe relación (negativos o −) o ambos (+/−). Format: formato del corpus.*

Como puede apreciarse en la tabla 4.1, muchos de los corpora contienen relaciones entre proteínas o anotaciones sintácticas, pero sólo BioInfer y LLL05 contienen ambos tipos de anotaciones. Un análisis detallado de la información contenida en esta tabla nos permite destacar las siguientes características:

- Existe una enorme variedad de esquemas de anotación, es decir, los corpus anotan distinta información y a distinto nivel. Por ejemplo, en algunos sólo se marcan las proteínas que interactúan, en otros además se dan palabras clave de la interacción, en otros el tipo de relación, etc. (para mayor detalle consultar tabla 4.2).
- No todos los corpora proporcionan ejemplos negativos (frases en las que aparecen menciones de proteínas pero no existe relación entre ellas). Este tipo de ejemplos son muy útiles para entrenar y validar cualquier sistema de detección de relaciones.
- Los distintos corpora utilizan formatos muy diversos. Los formatos más convenientes para este tipo de datos son los denominados *stand-off* (al margen del texto), que marcan la información de interés sin incluir etiquetas ni marcas en el texto original, sino que las marcas se definen aparte, referenciando posiciones del texto.
- La mayoría de los corpora son pequeños. Especialmente BioText, BioIE, Yapex, BioContrasts, Brown GENIA o LLL05, que apenas incluyen ejemplos suficientes para entrenar una herramienta potente, con capacidad de generalización adecuada para proprocionar un buen rendimiento en problemas reales.
- Sólo Wisconsin anota relaciones Gen-Enfermedad, pero este corpus sólo especifica si hay o no relación y cuáles son las entidades relacionadas, sin especificar el tipo de interacción ni marcar palabras clave o dependencias sintácticas (ver tabla 4.2).

En resumen, podemos afirmar que los corpora disponibles anotan información muy diversa y de forma muy heterogénea, por lo que la combinación de corpora resulta altamente compleja. Algunos trabajos recientes investigan en esta dirección, proponiendo modelos de consenso para las anotaciones sobre genes y proteínas de los corpora GENIA, GENETAG y AIMED [326].

La tabla 4.2 muestra un análisis más detallado de los corpora que contienen anotaciones para relaciones entre proteínas. Una inspección cuidadosa de esta tabla nos

Corpora	1.annotated proteins	2.interacting proteins	3.exact mention	4.keywords	5.interaction type	6.role arguments
BioText	*interacting proteins*	✓	×	×	✓	×
Wisconsin	*all*	✓	✓	×	×	×
PICorpus	*interacting proteins*	×	×	✓	×	×
Fetch Prot Corpus	*interacting proteins*	✓	×	×	×	×
HIV-1 HUMAN PI	*interacting proteins*	✓	×	×	✓	×
BioCreAtIvE I- PPI	*all*	✓	✓	✓	✓	✓
SPIES Corpus	*all*	×	×	×	×	×
BioIE	*interacting proteins*	✓	✓	✓	×	×
BioContrasts	*interacting proteins*	×	×	×	×	×
AIMED	*all*	✓	✓	×	×	×
BioInfer	*all*	✓	✓	✓	✓	✓
LLL05	*interacting proteins*	✓	✓	×	✓	✓
GENIA events	*all*	✓	✓	✓	✓	✓
BioNotate	*interacting proteins*	✓	✓	✓	×	×

Cuadro 4.2: *Características de los corpora con anotaciones para relaciones Proteína-Proteína. 1.- ¿Qué proteínas se anotan? (todas —all o sólo las que interactúan —interacting proteins). 2.- ¿Se marcan claramente qué proteínas interactúan?. 3.-En caso de que una proteína aparezca varias veces en el texto, ¿se marca claramente qué ocurrencia de la proteína interviene en la frase que las relaciona?. 4.- ¿Se proporcionan las palabras clave de la interacción?. 5.- ¿Se proporciona el tipo de interacción?. 6.- ¿Se proporciona el papel de las proteínas en la interacción?*

muestra que, de nuevo, la mayoría de los corpora no proporcionan suficiente información para recuperar las menciones exactas de las proteínas que interactúan y las palabras clave que soportan la interacción. Estas dos anotaciones son fundamentales para el entrenamiento de cualquier herramienta de extracción de relaciones basada en patrones o análisis sintáctico [156]. Aunque BioCreAtIvE I- PPI, BioIE, BioInfer y GENIA *events* satisfacen estos requisitos, el tamaño limitado de estos corpora hace necesario el desarrollo de nuevos esfuerzos y herramientas de anotación para proveer a la comunidad científica de un extenso corpus bien anotado para la identificación de relaciones entre entidades biomédicas de interés.

Esta revisión de los corpora existentes confirma la necesidad de crear un nuevo corpus siguiendo un esquema de anotación conveniente y con el suficiente esfuerzo de anotación para dotarlo de un número considerable de ejemplos, tanto positivos como negativos, que permita entrenar una herramienta de aprendizaje automático con garantías.

4.3 Herramientas de anotación de textos

En los últimos años han sugido diversas herramientas de anotación de textos de propósito general, como Knowtator [11], WordFreak [14], SAFE-GATE [79] o iAnnotate [10]. Estas herramientas proporcionan al usuario de mecanismos flexibles para definir el esquema de anotación, por lo que pueden ser adaptadas a la anotación de relaciones entre entidades biomédicas en textos científicos. Algunos grupos de BioNLP han creado sus propias herramientas *ad-hoc* para sus tareas de anotación de interés. Por ejemplo, Xconc Suite es una herramienta para anotar eventos en el corpus GENIA [174].

Estas herramientas están fundamentalmente diseñadas para esfuerzos de anotación que involucran a un pequeño número de anotadores bien entrenados, que realizan tareas de anotación muy específicas y conforme a esquemas de anotación sofisticados.

Sin embargo, en los últimos tiempos, con el auge de Internet y las comunidades virtuales, los proyectos colaborativos a gran escala están recibiendo una mayor difusión, aceptación y apoyo. Este tipo de esfuerzos muestran un enorme potencial, ya que permiten a millones de usuarios de todo el mundo colaborar en el desarrollo de un proyecto determinado, en este caso, la anotación de textos biomédicos. Las herramientas anteriores no están diseñadas para soportar este tipo de esfuerzos distribuidos de anotación a gran escala.

Existen herramientas de anotación colaborativa creadas específicamente para el campo de la biomedicina, como WikiGene [211], CBioC [41] y WikiProteins [229]. WikiGene y WikiProteins son dos entornos colaborativos basados en wikis. Los usuarios pueden editar las páginas asociadas con las entidades de interes y compartir su conocimiento acerca de las mismas con la comunidad. WikiGene está enfocado a genes y procesos de regulación genética. WikiProteins es un esfuerzo más ambicioso que permite la anotación de muchos tipos distintos de entidades (genes, proteinas, medicamentos, tejidos, enfermedades, etc.) y sus relaciones con otras entidades (creando lo que los autores denominan *Knowlets*).

Mientras que estos esfuerzos proporcionan a la comunidad con un medio para acceder y compartir gran cantidad de información indexada por entidades biológicas de interés, no están diseñados para servir en la creación de un corpus de texto que explícitamente constate la relación existente entre dichas entidades.

Nuestro enfoque está inspirado en esfuerzos colaborativos que recientemente han surgido en el campo del análisis y anotación de imágenes. Por ejemplo, *Image Labeler* de Google™(http://images.google.com/imagelabeler/) es una herramienta que permite etiquetar imágenes para mejorar los resultados de la búsqueda. Este sistema empareja aleatoriamente a dos usuarios y muestra a ambos usuarios el mismo conjunto de imágenes, pidiendo que asignen a cada una de las imágenes tantas etiquetas (y tan específicas) como sea posible durante un periodo limitado de tiempo. Cuando finaliza el tiempo se comparan las etiquetas proporcionadas por cada usuario, y en función del grado de coincidencia entre las respuestas y la especificidad de las etiquetas, se les otorga una puntuación (que dependerá de la calidad del etiquetado). Otro esfuerzo destacable en esta línea es *LabelMe* [265] (http://labelme.csail.mit.edu/), una aplicación desarrollada en el *MIT* para anotar contornos de objetos en imágenes. Esta aplicación implementa una interfaz en la que el usuario puede marcar siluetas en una imagen y asignarles una etiqueta con el nombre del objeto representado por esa silueta en la imagen. La Figura 4.2 muestra esta aplicación.

En la línea de este tipo de esfuerzos colaborativos, nosotros proponemos *BioNotate*, un esfuerzo global para la anotación de extractos de textos biomédicos con el objetivo de detectar relaciones entre entidades biomédicas de interés, como por ejemplo gen-gen (o proteína-proteína) y gen-enfermedad.

Nuestra propuesta es similar a la implementada por CBioC. Esta herramienta permite que el usuario anote relaciones entre conceptos biomédicos mientras navega a través de entradas de PubMed. Su funcionamiento se describe a continuación. Mientras el usuario visita abstracts de PubMed, se le muestran relaciones potenciales en-

Figura 4.2: *Captura de pantalla de la aplicación LabelMe [265] http://labelme.csail.mit.edu/, un sistema colaborativo de anotación de contornos de objetos en imágenes.*

tre entidades que han sido extraídas por sistemas automáticos o sugeridas por otros usuarios en base al texto de los abstracts. Usuarios registrados pueden añadir nuevas relaciones y confirmar o desmentir relaciones sugeridas por el sistema. La relación entre dos entidades se define proporcionando las dos entidades que interactúan y las palabras claves que soportan la interacción. Sin embargo, CBioC no permite proporcionar las menciones exactas de las entidades que intervienen en la relación. Además, el corpus completo de anotaciones no está directamente disponible para la comunidad científica, y sólo se puede acceder a las relaciones anotadas cuando se navega por registros de PubMed y se utiliza esta aplicación.

4.4 BioNotate: una herramienta de anotación colaborativa para textos biomédicos

La revisión de los corpora disponibles, realizada en la sección anterior, avala la necesidad de un nuevo paradigma comunitario para la anotación de corpora biomédico a gran escala y siguiendo esquemas de anotación adecuados. En esta sección proponemos BioNotate, una herramienta web de código libre que proporciona una plataforma de anotación colaborativa para textos biomédicos. Aunque el sistema originariamente ha sido concebido para la anotación manual de relaciones gen-gen y gen-enfermedad, recientes extensiones del mismo permiten implementar cualquier esquema de anotación (4.6.1), además de utilizar herramientas automáticas para asistir en la anotación manual y protocolos de la Web Semántica para exportar las anotaciones a otros recursos (4.6.3).

En esta sección se presenta la herramienta de anotación original sobre la que se desarrollan las extensiones descritas. Presentamos BioNotate como un sistema de anotación colaborativo que permite anotar pequeños extractos de texto (denominados comunmente *snippets*) de publicaciones biomédicas para identificar, en este caso, relaciones gen-gen y gen-enfermedad. La descripción de BioNotate se aborda en dos etapas:

1 Diseño del esquema y proceso de anotación.

2 Diseño y desarrollo de la herramienta de anotación.

4.4.1 Esquema y proceso de anotación

Para ilustrar la tarea de anotación que se pretende modelar, la tabla 4.3 incluye algunos extractos de texto reales con menciones a genes, proteínas y enfermedades. El primer extracto constata la existencia de una relación entre las proteínas *SCPA* y *C5a*, siendo el término *inhibits* el que evidencia dicha relación. Además, se nos informa de que se trata de una relación de tipo represivo, en el que *SCPA* actúa como agente represor y *C5a* padece esa represión. El segundo extracto explícitamente refuta la existencia de una relación entre el gen *APOE* y la enfermedad *autism*. El tercer extracto es un claro ejemplo de co-ocurrencia de dos entidades biomédicas de interés, en este caso las proteínas *D2* y *D1*, sin que en el texto se establezca una relación directa entre las mismas.

Siguiendo los ejemplos anteriores, proponemos un sistema para anotar pequeños extractos de texto (*snippets*) tomados de artículos biomédicos en los que existen

1	The action of SCPA enzymatically inhibits the chemotactic activity of C5a by cleaving its neutrophil binding site. [PMID: 12964111]
2	Three promoter, one intronic, and one 3' UTR single nucleotide polymorphisms (SNPs) in the APOE gene [...] as well as the APOE functional polymorphism (E2, E3, E4) were examined and failed to reveal significant evidence that autism is associated with APOE. [PMID: 14755445]
3	While stimulation of the D2 receptor increased branching and extension of neurites, stimulation of the D1 receptor reduced neurite outgrowth, suggesting that hormones and neurotransmitters may be capable of controlling the development of specific types of neurones.

Cuadro 4.3: *Extractos de textos biomédicos reales con menciones a genes, proteínas y enfermedades.*

menciones de distintos genes, proteínas o enfermedades y que son, por lo tanto, susceptibles de contener una relación entre estas entidades de interés. La idea central del esquema de anotación que proponemos consiste en mostrar un snippet al anotador y centrar su atención en una pareja de entidades que ha sido previamente marcada en el texto del snippet. En este contexto, pedimos que el anotador responda a la siguiente pregunta:

"*does this snippet imply a direct interaction between the provided entities?*".
("*¿Evidencia este extracto de texto la existencia de una relación directa entre las entidades marcadas?*").

La respuesta (Si/No) a esta pregunta nos permite clasificar los snippets en positivos (contienen una interacción) o negativos (no contienen una interacción). Sin embargo, para enriquecer la anotación del corpus y entrenar con garantías un sistema de text-mining, necesitaremos extraer más información de cada extracto de texto, en particular es fundamental conocer cuáles son las palabras que evidencian la existencia o no de una relación entre las entidades presentes en el texto. Por ejemplo, consideremos de nuevo los snippets de la Tabla 4.3. El primer snippet constata la existencia de una relación entre los genes *SCPA* y *C5a*. Entonces, el anotador respondería *Si* a la pregunta anterior, y marcaría *inhibits* como la palabra que evidencia la existencia de dicha interacción. El segundo snippet expresa una evidencia negativa de que el gen *APOE* está asociado con *autism*, pero mientras que estas entidades (*APOE* y *autism*) están sintácticamente conectadas por las palabras *is associated with*, el extracto que proporciona la semántica o mensaje del texto es *failed to reveal*. Entonces, el anotador debe marcar *failed to reveal* como la frase que soporta la "No" existencia, en

este caso, de una relación entre las entidades de interés. Es de destacar que el centro de atención de nuestra aproximación recae en la semántica o mensaje del texto, más que en la sintaxis del mismo.

Si las entidades simplemente co-ocurren en el texto del snippet sin que exista ninguna relación entre las mismas, como ocurre por ejemplo en el tercer snippet de la Tabla 4.3, la respuesta del anotador a la pregunta anterior debe ser *No* (las dos entidades no están relacionadas). Dado que no existe una secuencia de palabras en el texto que directamente evidencie la existencia o no de la relación, el anotador no debe marcar nada para justificar su respuesta en este caso. La anotación completa de los snippets de la Tabla 4.3 se muestra en la Tabla 4.4.

Consideramos que estas dos sencillas anotaciones: la respuesta Si/No a la pregunta anterior y el texto que evidencia la existencia o no de la interacción, constutiyen el conocimiento de mayor interés que un anotador humano puede proporcionar para la identificación de relaciones entre entidades biomédicas. Además, este protocolo es suficientemente sencillo e intuitivo para poder ser incrustado en una herramienta de anotación abierta a la comunidad.

Definición de Snippet

Para nuestro proyecto de anotación, definimos un snippet como un pequeño extracto de texto candidato a confirmar o descartar una relación entre dos entidades conocidas (genes o enfermedades). En particular, estamos interesados en dos tipos de snippets:

A) Aquellos que constatan o descartan una interacción entre un gen y una enfermedad (gen-enfermedad).

B) Aquellos que constatan o descartan una interacción entre dos genes o dos proteínas (gen-gen).

Para dar consistencia a nuestro proyecto, adoptaremos el convenio de que sólo existe una “interacción” o “relacion”, tanto positiva como negativa, entre dos entidades que co-ocurren en el texto de un snippet, si existe un extracto de dicho texto que soporta o descarta, explícitamente, la existencia de dicha relación.

Del mismo modo, sólo interesan las interacciones directas entre las dos entidades proporcionadas, lo que significa que frases como las siguientes:

- *Gene X regulates both A and B*
- *A and B play a role in autism*
- *A regulates the expression of X. X is associated to B*

no implican que exista una interacción directa entre *A* y *B*.

Proceso de Anotación

El proceso de anotación comienza mostrando al anotador el texto de un snippet y dos entidades de interés (gen-gen o gen-enfermedad) que aparecen en dicho texto. Una mención de cada entidad de interés es marcada *a priori* en el texto, para facilitar su localización y centrar la atención del usuario sobre las mismas. En este contexto, se pide que el anotador:

1 Indique Si/No, en función de si el texto implica que existe una relación directa entre las entidades de interés.

2 Marque la secuencia de palabras mínima y más significativa que justifique la respuesta anterior (si existe). Este texto se etiqueta como INTERACTION.

3 Localice y marque en el texto la mención de cada una de las entidades de interés que es esencial para expresar la relación entre las mismas (en su caso). Definimos estas menciones como aquellas que, si fueran omitidas o reemplazadas por menciones de otras entidades, modificarían el mensaje transmitido en el texto de forma que no se expresaría la misma relación. Por ejemplo, en el snippet:

```
Gene: Protein A
Gene: Protein B
Snippet: Protein A is found in tissue T. Protein A interacts with
protein B in the presence of catalyst C to produce D.
```

el cambio de la primera mención de *Protein A* por *protein E* no alteraría la relación que se expresa en el texto, mientras que el cambio de la segunda mención de *Protein A* por *protein E* si alteraría esta relación. Por tanto, el anotador debe marcar la segunda mención de *Protein A* junto con la mención de *protein B*.

Del mismo modo, en caso de que un pronombre haga referencia a la entidad de interés y participe en la frase que expresa la interacción, el anotador debe marcar el pronombre y no una mención de la entidad con la etiqueta correspondiente (GENE o DISEASE). Por ejemplo, en el snippet:

```
Gene: RELN
Disease: Autism
Snippet: Gene RELN was studied in various disorders. It turned out to
be causing autism.
```

It debe ser marcado como GENE porque hace referencia a uno de los genes de interés *RELN* y el cambio de este pronombre por otra entidad modificaría la relación expresada en el texto entre este gen y la otra entidad de interés: *autism*.

Esto también se aplica a frases nominales que se refieren a una de las entidades de interés, por ejemplo en el snippet:

```
Gene: FXR1
Gene: FMRP
Snippet: Recently, two proteins homologous to FMRP were discovered:
FXR1 and FXR2. These novel proteins interact with FMRP and with each
other. (PubMed 009259278)
```

El anotador debe marcar "These novel proteins" como un gen ya que hace referencia a uno de los genes de interés ("FXR1") y el reemplazo de esta frase nominal por una mención a otra entidad modificaría la relación expresada en el texto.

Sólo una mención de cada entidad de interés debe ser marcada en cada snippet. El anotador debe comprobar si las regiones marcadas satisfacen estas directrices y corregir las anotaciones que no lo hagan.

Etiquetas para anotación

El conjunto de etiquetas o marcas disponible para el proceso de anotación es el siguiente:

- GENE: para menciones de genes o proteínas, por ejemplo: RELN, GRM8, WNT2, etc.
- DISEASE: para menciones de enfermedades, por ejemplo: autistic disorder, AutD, ASD, etc.
- INTERACTION: secuencia mínima y más importante de palabras que justifican la existencia o no de una interacción, por ejemplo: "binds to", "phosphorylates", etc.

La anotación completa de los snippets de la Tabla 4.3 se muestra en la Tabla 4.4.

Instrucciones detalladas para los anotadores y más ejemplos anotados están a disposición de los anotadores en el sitio web de BioNotate.

1	Does this snippet imply a direct interaction between the provided entities?: Yes. `The action of SCPA enzymatically inhibits the chemotactic activity of C5a by cleaving its neutrophil binding site. [PMID: 12964111]`
2	Does this snippet imply a direct interaction between the provided entities?: No `Three promoter, one intronic, and one 3' UTR single nucleotide polymorphisms (SNPs) in the APOE gene [...] as well as the APOE functional polymorphism (E2, E3, E4) were examined and failed to reveal significant evidence that autism is associated with APOE. [PMID: 14755445]`
3	Does this snippet imply a direct interaction between the provided entities?: No `While stimulation of the D2 receptor increased branching and extension of neurites, stimulation of the D1 receptor reduced neurite outgrowth, suggesting that hormones and neurotransmitters may be capable of controlling the development of specific types of neurones.`

Cuadro 4.4: *Ejemplos de snippets anotados. En verde se indican los tokens etiquetados como* GENE *o* DISEASE *y en amarillo como* INTERACTION*. Nótese que en el snippet (3) las dos entidades de interés (D1 y D2) co-ocurren sin que el mensaje del texto evidencie la existencia de una relación entre ellas. Por tanto, la respuesta es 'No' y el anotador no debe marcar nada como* INTERACTION *en este caso.*

4.4.2 Sistema de Anotación

En esta sección abordamos el diseño e implementación de la herramienta de anotación que de soporte al esquema y procesos de anotación comentados en la sección anterior. Las características fundamentales de este sistema de anotación son las siguientes:

- Simultaneidad de las anotaciones. La tarea de anotación debe desarrollarse en paralelo por varios anotadores diferentes.
- Gestión de anotadores. El sistema debe registrar todas las anotaciones efectuadas por cada usuario. Del mismo modo, también deben permitirse anotaciones anónimas.
- Política de distribución de los snippets entre los anotadores. Cuando un anotador solicita un nuevo snippet para anotar, el sistema le asigna un nuevo snippet de un conjunto de snippets *pendientes de anotación*. Un mismo snippet no será nunca mostrado a un usuario registrado si éste ya lo ha anotado previamente. Para asegurar la calidad del corpus resultante, requerimos que al menos k anotaciones realizadas por usuarios diferentes alcancen un mínimo grado de acuerdo para cada snippet. La Sección 4.4.2.1 detalla este proceso.
- Acceso al sistema de anotación desde cualquier máquina con conexión a internet, sin necesidad de la instalación previa de un software. Nuestro propósito es concentrar el esfuerzo del anotador en la tarea de anotación y permitirles

contribuir con nuevas anotaciones desde cualquier máchina y en cualquier momento.

- Software *open-source* y de libre distribución. Nuestro objetivo es permitir el desarrollo de multitud de pequeños esfuerzos en paralelo que se adapten a las necesidades del usuario. Por ello proporcionamos el código de nuestro software para que otros grupos puedan descargarlo, adaptarlo a sus necesidades (implementar su propio esquema de anotación) y desarrollar sus propios esfuerzos colaborativos de anotación, compartiendo con la comunidad, si así lo desean, el corpus anotado resultante.

Tras el análisis detallado de estos requisitos y directrices de diseño, decidimos implementar el sistema de anotación colaborativa como un sistema web, de forma que los usuarios no necesiten realizar ninguna instalación de software, y puedan realizar anotaciones en cualquier momento y desde cualquier máquina con acceso a la red.

4.4.2.1 Distribución de los snippets a los anotadores

Cuando un anotador accede al sistema y solicita la anotación de un nuevo snippet, el sistema le asigna un nuevo snippet de un conjunto de snippets *pendientes de anotación*. El snippet será escogido aleatoriamente dentro de este conjunto, comprobando que no ha sido previamente anotado por el mismo usuario. Cada snippet debe ser anotado por, al menos, k anotadores diferentes. Si las k anotaciones de un snippet no alcanzan un nivel mínimo de acuerdo, el snippet debe presentarse a otro anotador al azar. El proceso continúa hasta que al menos k anotaciones sobre un mismo snippet alcancen el mínimo nivel de acuerdo (Ver Figura 4.3).

Establecemos que k anotaciones sobre un snippet alcanzan un grado de acuerdo mínimo si satisfacen las siguientes condiciones:

1 La respuesta Si/No es la misma.

2 La secuencia de tokens marcada con las etiquetas GENE y/o DISEASE se solapan completamente.

3 La secuencia de tokens marcada con la etiqueta INTERACTION se solapan significativamente, permitiendo hasta 1 token diferente respecto a la secuencia más corta para cada una de las parejas de anotaciones de tipo INTERACTION de las k anotaciones.

Por ejemplo, consideremos el snippet (1) de la Tabla 4.3. Si *Anotador1* marca "inhibits" como INTERACTION y *Anotador2* marca "inhibits the activity of" con la misma

etiqueta, las dos anotaciones alcanzarían el mínimo grado de acuerdo para esta etiqueta ya que ninguno de los tokens de la secuencia marcada más corta ("inhibits") es diferente de los tokens de la secuencia más larga ("inhibits the activity of"). Si un nuevo anotador, *Anotador3*, marca "enzymatically inhibits", esta anotación estaría en acuerdo con lo anotado por *Anotador1* (por la misma razón que el anterior) y con *Anotador2*: sólo una palabra ("enzymatically") de la secuencia marcada más corta (la de *Anotador3*) no está incluida en la secuencia más larga (la de *Anotador2*). Si un nuevo anotador, *Anotador4*, marca "action of SCPA enzymatically inhibits" como INTERACTION, esta anotación no alcanzaría el mínimo grado de acuerdo con la anotación de *Anotador2*, pero sí alcanzaría el acuerdo con las anotaciones de *Anotador1* y *Anotador3*.

4.4.2.2 Características técnicas del sistema de anotación

BioNotate es una plataforma web cliente/servidor implementada en JavaScript. Del lado del cliente, la aplicación consta de una interfaz de usuario sencilla e intuitiva, donde se muestran los snippets y el usuario puede realizar anotaciones sobre ellos. Las anotaciones consisten en el marcado de secuencias de texto y la asignación de las correspondientes etiquetas conforme a lo especificado en 4.4.1. Una captura de pantalla de la aplicación se muestra en la Figura 4.6. Algunos aspectos técnicos sobre la interfaz se detallan a continuación:

- El usuario puede marcar cualquier extracto arbitrario de texto del snippet. Esta marcación puede sobrepasar marcas de formato HTML, por ejemplo empezar en un párrafo y acabar en otro.
- Existen dos tipos de anotaciones: genes (color verde) e interacción (color amarillo). Dos o más anotaciones pueden solaparse. Si se produjera solapamiento, el color resultante es una combinación de los colores combinados.
- El panel del margen derecho del texto del snippet registra las anotaciones efectuadas por el usuario en el snippet actual. Cada anotación tiene una entrada asociada en el panel. Desde este panel el usuario también puede borrar cualquier anotación.
- Un pequeño panel permite al usuario identificarse en cualquier momento. El ususario también puede efectuar las anotaciones de forma anónima. El sistema registra las anotaciones efectuadas por cada usuario.
- El usuario puede descartar la anotación de un snippet si el texto le resulta confuso o no está seguro de su respuesta.

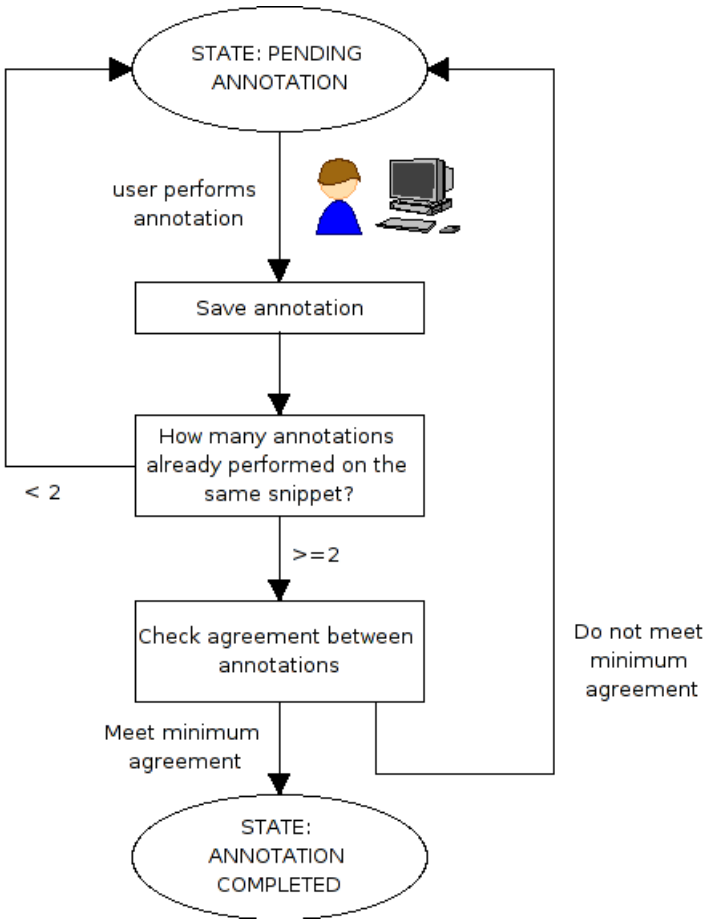


Figura 4.3: *Diagrama de estados representando el proceso de anotación de un snippet, desde el estado "pendiente de anotación" al estado "anotación completada". La anotación de un snippet se completa cuando éste ha sido anotado por, al menos, k usuarios diferentes y las anotaciones alcanzan un grado de acuerdo mínimo. Cuando el snippet alcanza el estado "anotación completada" no será servido de nuevo a ningún anotador.*

- El sistema es totalmente funcional en disntintos navegadores: *Firefox*, *Internet Explorer*, *Opera* y *Safari*.

La implementación del marcado de texto y la gestión de anotaciones en la interfaz requieren un uso continuado de Javascript y el *Data Object Model* (DOM) para la recuperación y actualización de componentes en la página de forma dinámica. Para la implementación del marcado de texto se ha reutilizado parte de código de la herramienta de software libre *Marginalia*[123].

Del lado del servidor, la aplicación consta de dos scripts implementados en Perl. El primero sirve snippets que no han sido anotados por el usuario que realiza la petición. El segundo guarda las anotaciones de un usuario y comprueba si alcanzan un mínimo

grado de acuerdo con las de otros usuarios. Tanto los snippets sin anotar como las anotaciones se almacenan en ficheros en formato XML. Una lista de los snippets que están pendientes de anotación, junto con la información de qué usuarios los han anotado previamente, se almacena en un fichero de texto plano que los scripts de Perl leen y modifican conforme se añaden nuevas anotaciones. La comunicación entre el cliente y el servidor se implementa utilizando la tecnología AJAX (Asynchronous JavaScript and XML).

Más detalladamente, el sistema trabaja de la siguiente forma. Cuando un anotador entra en BioNotate, el código JavaScript ejecutándose en el cliente solicita del servidor un nuevo snippet para el usuario. En dicha solicitud se proporciona el nombre de usuario del anotador (o 'anonymous' si el anotador no se ha registrado). El primer script de Perl que se ejecuta en el servidor recibe esta solicitud y devuelve un snippet que no haya sido anotado por dicho usuario, si existe alguno disponible. Cuando el navegador del usuario recibe el snippet, lo carga en la ventana de anotación para ponerlo a disposición del usuario. Una vez que el usuario finaliza la anotación del snippet y confirma que desea guardar la anotación, el cliente manda este snippet anotado al servidor. El segundo script de Perl gestiona este snippet anotado, guardándolo en formato XML y confirmando si existe acuerdo o no con anotaciones realizadas por otros usuarios, tal y como se describe en la Sección 4.4.2.1. Un esquema de este proceso se muestra en la figura 4.4. Dado que los datos del usuario se guardan como parte de la anotación, los snippets anotados por un usuario particular pueden ser fácilmente recuperados a partir del corpus completo una vez que el mismo se hace disponible a la comunidad.

La Figura 4.7 muestra el flujo de información de entrada y salida de BioNotate. El sistema debe ser provisto de snippets en formato XML en los cuales dos entidades de interés han sido proporcionadas para cada uno de ellos. Las anotaciones resultantes se proporcionan también en XML. Un extracto de un snippet en formato XML se muestra en la Figura 4.5. El sistema también genera un fichero de texto plano con una lista de snippets para los que se ha alcanzado un mínimo grado de acuerdo (según los criterios descritos en 4.4.2.1) y las referencias a las anotaciones para las que se cumple dicho acuerdo. Todos los componentes de este sistema de anotación están disponibles en la página de BioNotate, así como documentación completa en la que se describe detalladamente el formato de los ficheros XML e instrucciones de configuración paso a paso.

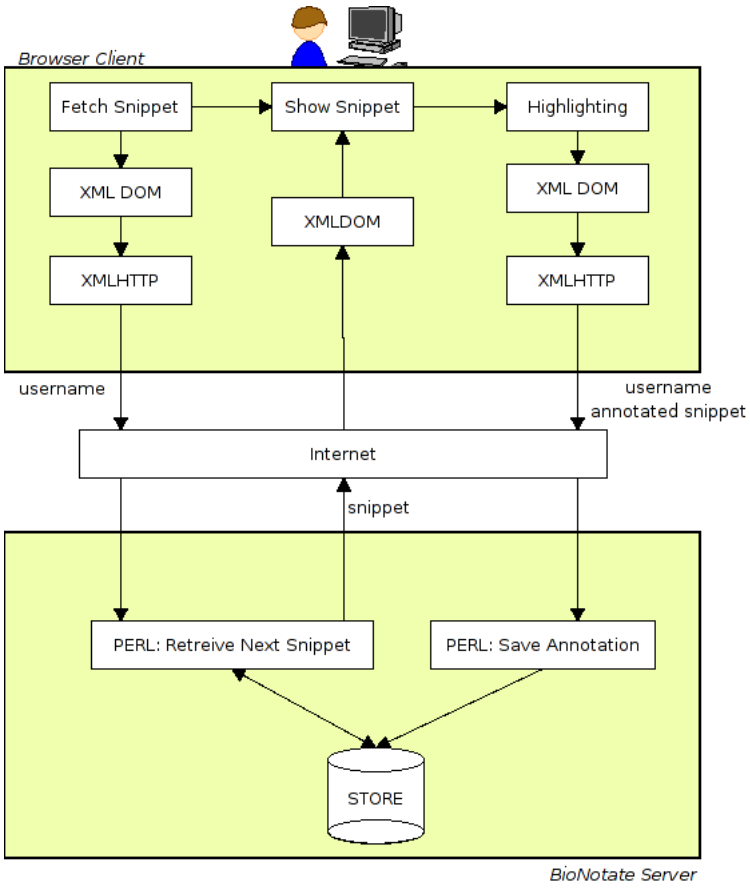


Figura 4.4: *Componentes y comunicaciones Cliente/Servidor en BioNotate. El cliente itera en el bucle: Obtener el siguiente snippet para el anotador actual - Presentarlo para que pueda realizar las anotaciones - Salvar el snippet. El proceso continúa hasta que el usuario cierra la ventana del navegador. Los módulos del cliente se comunican con el servidor para recuperar snippets sin anotar y guardar las anotaciones efectuadas por el usuario.*

4.5 Caso de estudio: corpus piloto sobre autismo

Como ejemplo de uso de BioNotate, esta sección presenta un esfuerzo piloto de anotación de un corpus con interacciones entre genes relacionados con autismo. En ella describimos los métodos para crear el corpus y presentamos los primeros resultados del esfuerzo de anotación. Este trabajo se ha realizado en colaboración con el Centro de Informática BioMédica de la Universidad de Harvard.

4.5.1 Motivación y Objetivos.

El Autismo (*Autism Spectrum Disorder*) es una enfermedad multigenética compleja con un amplio espectro de fenotipos. Aunque está claro que el autismo es hereditario,

```
– <snippetID>
    PubMed_015632144_9606.ENSP00000329120_9606.ENSP00000327160_2
  </snippetID>
– <source>
    <name>PubMed</name>
    <sourceId>015632144</sourceId>
  </source>
– <location>
    ./annotations/STRING_CORPORA
    /PubMed_015632144_9606.ENSP00000329120_9606.ENSP00000327160_2.xml
  </location>
  <author>anonymous</author>
– <text>
    Disabled-1 ( Dab1 ) is an essential adaptor protein that functions in the Reelin signaling
    pathway and is required for the regulation of neuronal migration during embryonic
    development . Dab1 interacts with NPXY motifs in the cytoplasmic tails of the lipoprotein
    receptors ApoER2 and very low density lipoprotein receptor through an amino-terminal
    phosphotyrosine binding ( PTB ) domain (2).
  </text>
– <EoIs>
  – <EoI>
      <id>1</id>
      <symbol>Dab1</symbol>
      <type>gene</type>
      <officialId>ENSP00000329120</officialId>
      <officialIdType>Ensembl</officialIdType>
    </EoI>
```


Figura 4.5: *Extracto de un snippet en formato XML.*

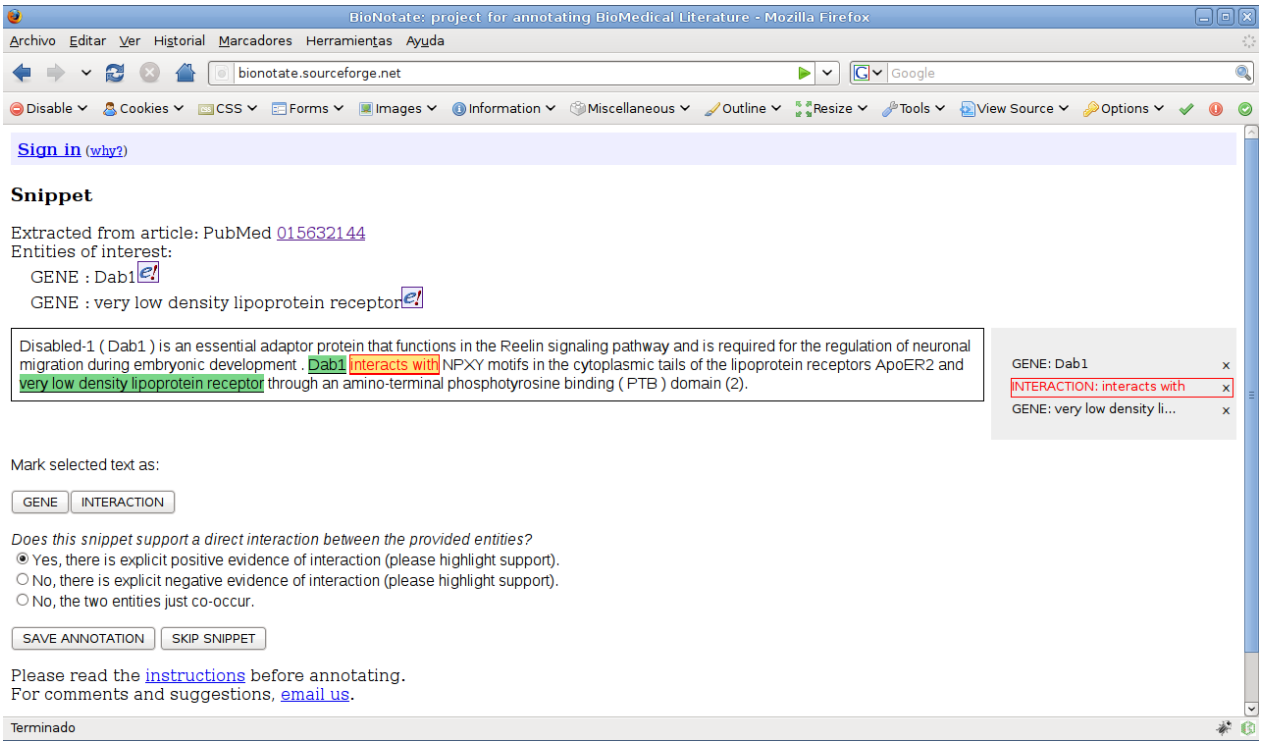


Figura 4.6: *Captura de pantalla del interfaz de anotación de BioNotate. La imagen muestra un snippet en el que dos entidades y una relación han sido marcadas.*

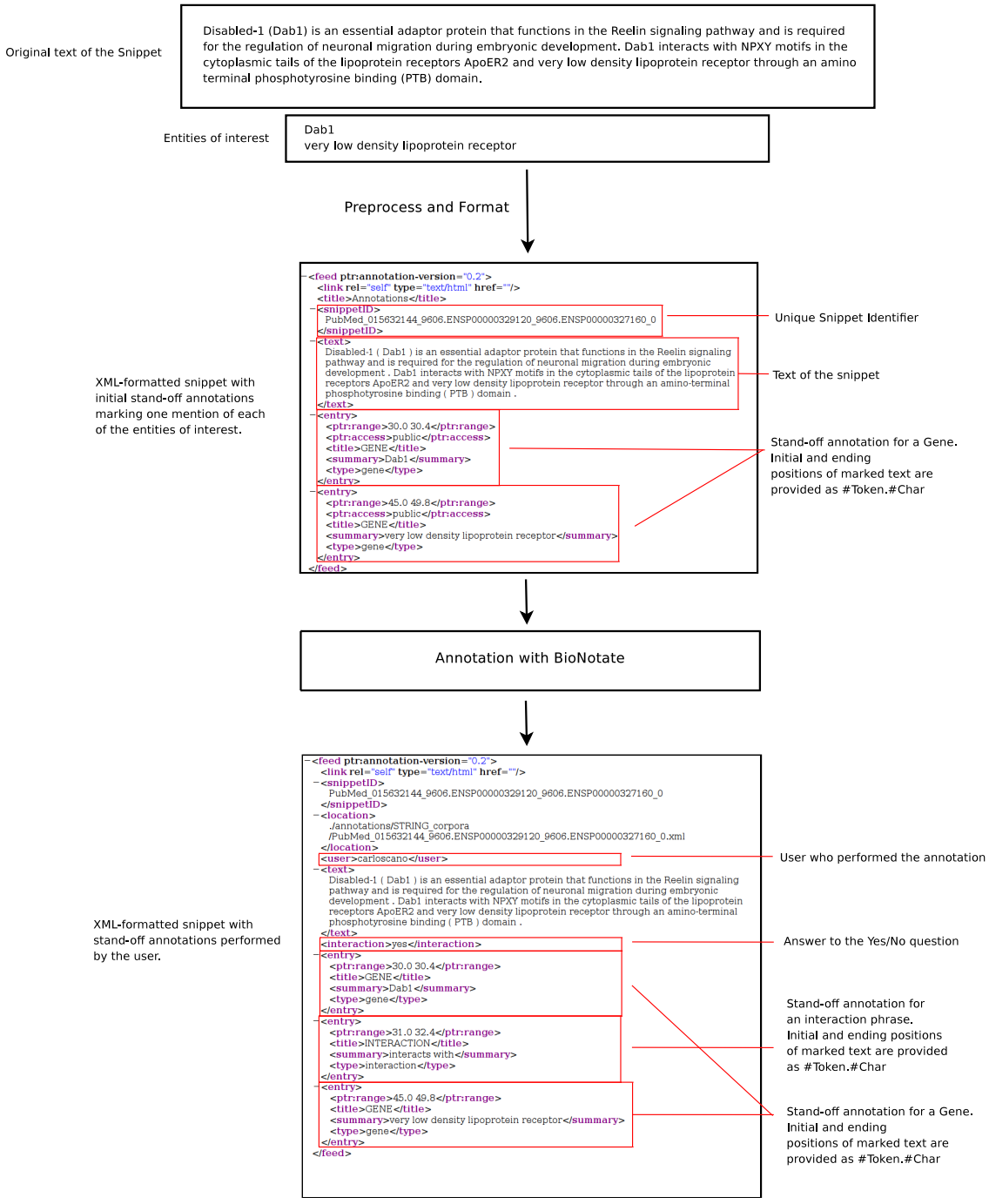


Figura 4.7: *Flujo de información de entrada y salida de BioNotate.*

la identificación de los agentes moleculares que lo causan siguen siendo una incógnita, y no está claro si el componente genético del autismo es una combinación de unos pocos variantes comunes o de multitud de variantes poco frecuentes. Hasta la fecha, más de 100 genes han sido relacionados con el autismo en la literatura, cada uno de los cuales está involucrado en numerosos procesos biológicos distintos y presenta diferentes interacciones moleculares. Es por esto que el estudio de autismo está evolucionando del análisis individualizado de genes al análisis de complejas redes genéticas.

En este contexto, el Centro de Informática BioMédica de la Universidad de Harvard, está construyendo una red genética completa para el autismo, en la que se representan los genes relacionados con esta enfermedad y sus interacciones moleculares, en un proyecto denominado Autworks (http://autworks.hms.harvard.edu). Los nodos de esta red genética representan genes involucrados en Autismo y los arcos relaciones entre los mismos, extraídas de distintas fuentes de datos biológicas, entre ellas, de la literatura biomédica.

Nuestro objetivo para este caso de estudio es la validación de las relaciones entre genes extraídas de la literatura mediante bases de datos y herramientas existentes y representadas en la red genética del proyecto Autworks.

4.5.2 Fuentes de datos y métodos para la creación del corpus.

La fuente principal de datos es PubMed, un motor de búsqueda de uso muy extendido entre la comunidad científica que permite la búsqueda sobre más de 19 millones de artículos científicos almacenados en su base de datos: MEDLINE. Como se ha expuesto anteriormente, la búsqueda en PubMed de genes, proteínas o enfermedades cuyo estudio está medianamente extendido, devuelve una enorme cantidad de resultados. Para limitar la búsqueda a artículos que reporten relaciones gen-gen (proteína-proteína) y gen-enfermedad, el proyecto Autworks utiliza herramientas y bases de datos accesibles al público como STRING [322].

STRING (ver Sección 1.3.1.3 para una descripción más detallada) es una base de datos de interacciones proteína-proteína basada fundamentalemente en la co-ocurrencia significativa de las proteínas en el texto de las publicaciones. La entrada es uno o más nombres de proteínas, para los que STRING devuelve un grafo donde los nodos representan las proteínas y los arcos las relaciones entre ellas encontradas en la literatura. Para cada arco de este grafo, STRING proporciona la lista de publicaciones que sustenta esta relación.

Para obtener el corpus piloto a partir de la red genética del autismo del proyecto Autworks, procedemos del siguiente modo:

1 Consultamos OMIM [12] y GeneCards [6] para recuperar genes y proteínas asociados a la palabra clave “autism”.

2 Consultamos STRING[1] para recuperar todas las publicaciones que soportan una relación entre cada par de proteínas (genes) de la lista obtenida en el paso anterior. Estas publicaciones constituyen la evidencia textual asociada a los arcos de la red genética de autismo en Autworks. Específicamente, la información obtenida de STRING para cada una de estas publicaciones es el PubMedID, el ID de Ensembl para las dos proteínas de interés, el texto del abstract y todas las menciones de las proteínas en el texto. Como STRING ya lleva a cabo una búsqueda e identificación de las entidades de interés en el texto, no se utiliza ningún método adicional de *Named Entity Recognition (NER)*.

3 Para cada abstract que soporte una relación entre cada pareja de proteínas, extraemos todos los extractos de texto interesantes (snippets) que son candidatos a respaldar dicha interacción. El proceso de extracción de snippets del texto se detalla en la sección 4.5.2.1. Los snippets extraídos constituyen el corpus piloto.

El corpus resultante contiene snippets candidatos a respaldar 168 relaciones entre 127 proteínas. Se procesaron un total de 2,053 abstracts generándose un total de 1,819 snippets.

4.5.2.1 Extracción de snippets del texto

Una vez que todas las menciones de las proteínas (genes) de interés en un texto han sido identificadas (en nuestro caso, STRING incorpora su propio sistema NER), creamos un snippet por cada par de ocurrencias de distintas proteínas que estén suficientemente cercanos en el texto. Cada snippet contendrá el texto entre las ocurrencias de las dos entidades de interés, y algo de texto anterior a la primera ocurrencia y posterior a la última, para proporcionar al anotador cierto contexto que le ponga en situación.

El esquema siguiente describe los pasos del algoritmo:

1 Recuperar dos entidades de interés: X,Y que estén a distancia máxima de MAX_LENGHT_CORE tokens.

2 Todo el texto entre X e Y se incluye en SNIPPET.

[1] STRING version 6,3

3 Extender SNIPPET de X hacia atrás hasta el inicio de la sentencia que contiene a X, y de Y hacia delante hasta el fin de la sentencia que contiene Y, siempre y cuando el tamaño total del SNIPPET no sobrepase MAX_LENGHT_TOTAL tokens.

4 Si la longitud del SNIPPET es inferior a MIN_LENGHT_TOTAL, extender SNIPPET hacia atrás incluyendo la frase anterior, y hacia delante incluyendo la frase posterior, siempre y cuando no se exceda el límite de MAX_LENGHT_TOTAL tokens.

5 Devolver SNIPPET

Por lo tanto, la longitud máxima (en número de tokens) de un snippet es MAX_LENGHT_TOTAL, y MAX_LENGHT_CORE representa la distancia máxima entre las dos entidades de interés del snippet. También empleamos la constante MIN_LENGHT_TOTAL para garantizar que todos los snippets tienen suficiente contexto para que sean fácilmente inteligibles en el proceso de anotación. El establecimiento de valores para estas constantes depende del tipo de textos que analicemos. Hemos comprobado experimentalmente que una longitud total (MAX_LENGHT_TOTAL) de 300 tokens es apropiada para los snippets procedentes de textos biomédicos, con un MAX_LENGHT_CORE de 240 tokens y MIN_LENGHT_TOTAL de 40.

En el caso de que exista más de una mención de las entidades de interés en el texto del snippet, las dos menciones más cercanas se marcaran *a priori* para crear los snippets que se cargan en BioNotate.

El pseudocódigo detallado del proceso de extracción de snippets se muestra en Algoritmo 1.

4.5.3 Resultados sobre el corpus piloto.

Como prueba de concepto y para validar la metodología implementada por BioNotate, esta sección presenta los primeros resultados del esfuerzo piloto de anotación sobre autismo.

El corpus resultante consta, por una parte, de 1,000 snippets anotados por un único usuario. El análisis de este corpus y de las entidades anotadas permite extraer las primeras conclusiones de interés. La primera de estas conclusiones es que, de acuerdo con este único anotador, sólo 116 snippets del total de 1,000 analizados realmente evidencian la existencia de una relación entre las proteínas proporcionadas.

Algoritmo 1: *Creación de snippets a partir de un texto.*

for all $text$ in corpus **do**

$ListOccA \leftarrow$ Find all the occurrence of $GeneA$ in $text$
$ListOccB \leftarrow$ Find all the occurrence of $GeneB$ in $text$

for all $pair(OccA, OccB)$ in $ListOccA$, $ListOccB$ **do**
$Dist \leftarrow$ Compute distance between $OccA$ and $OccB$ in $text$
if $Dist <= MAX\_LENGTH\_CORE$ **then**
Add $(OccA, OccB)$ to $ListSnippets$
end if
end for

Remove every $(OccA, OccB)$ from $ListSnippets$ which fulfils:
$text(OccA, OccB) \subset text(OccAi, OccBi)$, with $(OccAi, OccBi) \in ListSnippets$

Combine every pair of snippets: $(OccA, OccB), (OccA', OccB')$ from $ListSnippets$ fulfilling:
$text(OccA, OccB)$ OVERLAPS with $text(OccA', OccB')$ AND
$Dist(OccA, OccB') <= MAX\_LENGTH\_CORE$ AND
$Dist(OccA', OccB) <= MAX\_LENGTH\_CORE$

for all snippet $(OccA, OccB) \in ListSnippets$ **do**

$SnippetText \leftarrow text(OccA, OccB)$
$FirstOcurrence \leftarrow FirstOccurrence(OccA, OccB)$
$LastOccurence \leftarrow LastOccurrence(OccA, OccB)$
$DistToStart \leftarrow ComputeDistance(FirstOccurrence, Start(sentence(FirstOccurrence)))$
$DistToEnd \leftarrow ComputeDistance(LastOccurrence, End(sentence(LastOccurrence)))$

if $DistToStart + DistToEnd <= MAX\_LENGTH - MAX\_LENGTH\_CORE$ **then**
Extend $SnippetText$ back to $Start(sentence(FirstOccurrence))$
Extend $SnippetText$ forward to $End(sentence(LastOccurrence))$
else
if $DistToStart <= MAX\_LEGTH\_BEGINNING$ **then**
Extend $SnippetText$ back to $Start(sentence(FirstOccurrence))$
Extend $SnippetText$ forward $(MAX\_LENGTH - MAX\_LENGTH\_CORE - DistToStart)$ tokens
else
Extend $SnippetText$ back $((MAX\_LENGTH - MAX\_LENGTH\_CORE)/2)$ tokens
Extend $SnippetText$ forward $((MAX\_LENGTH - MAX\_LENGTH\_CORE)/2)$ tokens
end if
end if

Save $SnippetText$ in $ListSnippetTexts$

end for
end for

Return $ListSnippetsTexts$

Esto implica que la tasa de error para los arcos de Autworks con soporte textual es de, aproximadamente, el 89 %.

Un análisis detallado de las entidades e interacciones marcadas por el usuario revela que, de las 200 entidades marcadas como GENE, no todas hacen referencia a entidades distintas, siendo común el uso de nombres sinónimos para referirse a un mismo gen, por ejemplo: VLDL-R, VLDLR, VLDLr o 5-HT-2A, 5-HT2A, 5HT2A. Respecto a las palabras marcadas con la etiqueta INTERACTION, encontramos muchos verbos de acción como, por ejemplo: "associated with", "docks to", "binds", "phosphorylated by". También observamos numerosos casos con incertidumbre como "probably unrelated genes", "may interact with" y "little is known about". Las secuencias de palabras marcadas como INTERACTION tienen una extensión promedio de 4 palabras, con un rango que va desde 1 a 28 palabras.

Por otro lado, y para evaluar el grado de acuerdo entre anotadores, efectuamos la anotación con cuatro anotadores de un corpus reducido. Para esta evaluación seleccionamos el conjunto de snippets para los que el primer anotador efectuó algún etiquetado para INTERACTION, es decir, seleccionamos aquellos snippets que, de acuerdo con dicho anotador, contienen evidencia explícita (bien sea positiva o negativa) de que existe una relación entre las entidades proporcionadas. Este subconjunto contiene 139 snippets. Para su anotación, involucramos a tres anotadores más y establecemos $k = 2$ como número mínimo de anotadores para los que sus anotaciones tienen que alcanzar un grado de acuerdo significativo, según los criterios presentados en la Sección 4.4.2.1.

La Tabla 4.5 muestra los resultados obtenidos en término de acuerdo entre anotadores. Anteriores esfuerzos de anotación para la identificación y normalización de nombres de genes arrojan tasas de acuerdo que varían entre el 91 % y el 69 % para ciertos contextos [75]. En nuestro caso, el porcentaje promedio de acuerdo por anotación es superior al 75 % y la tarea de anotación incluye la anotación de las entidades que interaccionan y las palabras clave que soportan la interacción. Por lo tanto, y dado que las tasas de acuerdo mostradas son similares a las de otros esfuerzos de anotación, podemos concluir que la aproximación que proponemos es efectiva para la anotación distribuida y colaborativa de textos.

El análisis de las anotaciones sin acuerdo entre anotadores revela algunos errores no intencionados introducidos por los anotadores en casos esporádicos. Por ejemplo, anotar que no existe relación pero marcar un extracto de texto que claramente implica una relación positiva entre las entidades de interés. Otro motivo más frecuente de

Anotador	N.Snippets	N.Snippets con acuerdo	% acuerdo
1	139	94	0.676
2	138	111	0.804
3	48	38	0.792
4	44	35	0.795
Total	369	278	0.753

Cuadro 4.5: *Número de snippets anotados (N.Snippets), número de snippets anotados con acuerdo y % del total de snippets con acuerdo por anotador, de acuerdo con el criterio presentado en la Sección 4.4.2.1. El tamaño del corpus es de 139 snippets.*

desacuerdo es la presencia de varias frases distintas que avalan la interacción entre dos proteínas en un mismo snippet, por ejemplo:

`The KH domains of FXR1 and FMR1 are almost identical, and the two proteins have similar RNA binding properties in vitro. However, FXR1 and FMR1 have very different carboxy-termini. [...] These findings demonstrate that FMR1 and FXR1 are members of a gene family and suggest a biological role for FXR1 that is related to that of FMR1.`

En algunas ocasiones, largas frases que avalan la interacción son también fuente de desacuerdo entre anotadores, por ejemplo:

`By immunoblotting , we found that a marked reduction in FMRP levels is associated with a modest increase in FXR1P (PubMed 012112448).`

`No association between the very low density lipoprotein receptor gene and late-onset Alzheimer's disease nor interaction with the apolipoprotein E gene in population-based and clinic samples . (PubMed 009181358)`

Otra fuente de desacuerdo es el marcado de pronombres y frases nominales que se refieren a una de las entidades de interés, de acuerdo con las directrices proporcionadas en la Sección 4.4.1. Por ejemplo, en la frase:

`The biological role of the very low density lipoprotein receptor ( VLDL-R ) in humans is not yet elucidated . This cellular receptor binds apolipoprotein E (apoE)-containing lipoparticles and is mainly expressed in peripheral tissues. (PubMed 009409253).`

En este caso, un anotador señaló la mención del gen "very low density lipoprotein receptor", mientras que otros dos anotadores marcaron la frase nominal "This cellular receptor", que se refiere a dicha entidad y cuyo reemplazo por otra entidad alteraría la relación que se expresa en el texto.

Dado que requerimos que exista un grado de acuerdo significativo entre al menos $k = 2$ anotadores, errores y desacuerdos de los anotadores se descartan y el corpus resultante muestra anotaciones de consenso y, por tanto, de mejor calidad. Este análisis detallado de las anotaciones efectuadas por cada usuario nos ha permitido mejorar las directrices de anotación y enriquecer la documentación con más ejemplos y más ilustrativos.

De acuerdo con el corpus resultante de este esfuerzo de anotación con varios anotadores, 110 snippets del total de 139 snippets contienen una relación positiva entre las entidades de interés, otros 8 contienen una relación explícita negativa y los restantes 19 no presentan ninguna relación. El 76 % de los snippets que contienen una relación, las entidades que interaccionan fueron aquellas marcadas *a priori* por el sistema (recordemos que las dos menciones más cercanas de las entidades de interés se marcaban *a priori* tal y como se describe en la Sección 4.5.2.1). Para el 18 % de los snippets con relación, las menciones que interaccionan no fueron aquellas marcadas *a priori*, pero el texto de las mismas si era igual que el proporcionado. En el restante 6 % las entidades que interaccionan eran sinónimos de aquellas que se proporcionaban de inicio (como 'methyl CpG binding protein 2', 'MECP2'), pronombres o frases nominales que se refieren a dichas entidades (como "This cellular receptor").

4.6 Extensiones y aplicaciones de BioNotate.

4.6.1 Flexibilización del esquema y proceso de anotación.

El esquema de anotación implementado por BioNotate (descrito en la Sección 4.4) resulta adecuado para la anotación de relaciones gen-gen (o proteína-proteína) y gen-enfermedad. Sin embargo, la identificación automática de este tipo de relaciones es sólo uno de los problemas de text-mining a los que se enfrenta la comunidad científica. La detección de nombres de genes, enfermedades, especies, SNPs, fármacos, tejidos, términos de ontologías biológicas (como la Gene Ontology) y otras muchas entidades de interés continúa siendo uno de los problemas de mayor trascendencia en el campo (ver sección 1.3.1.1). Además, el interés también recae en la detección de otro tipo de relaciones distintas de gene-gen o gen-enfermedad, como relaciones gen-SNP, fármaco-enfermedad, gen-SNP-enfermedad, etc. De este modo, y para extender el uso de BioNotate entre la comunidad científica, hemos implementado una nueva versión de la herramienta en la que el usuario puede definir cómodamente su propio esquema de anotación y la función que define cuándo hay acuerdo entre anotadores.

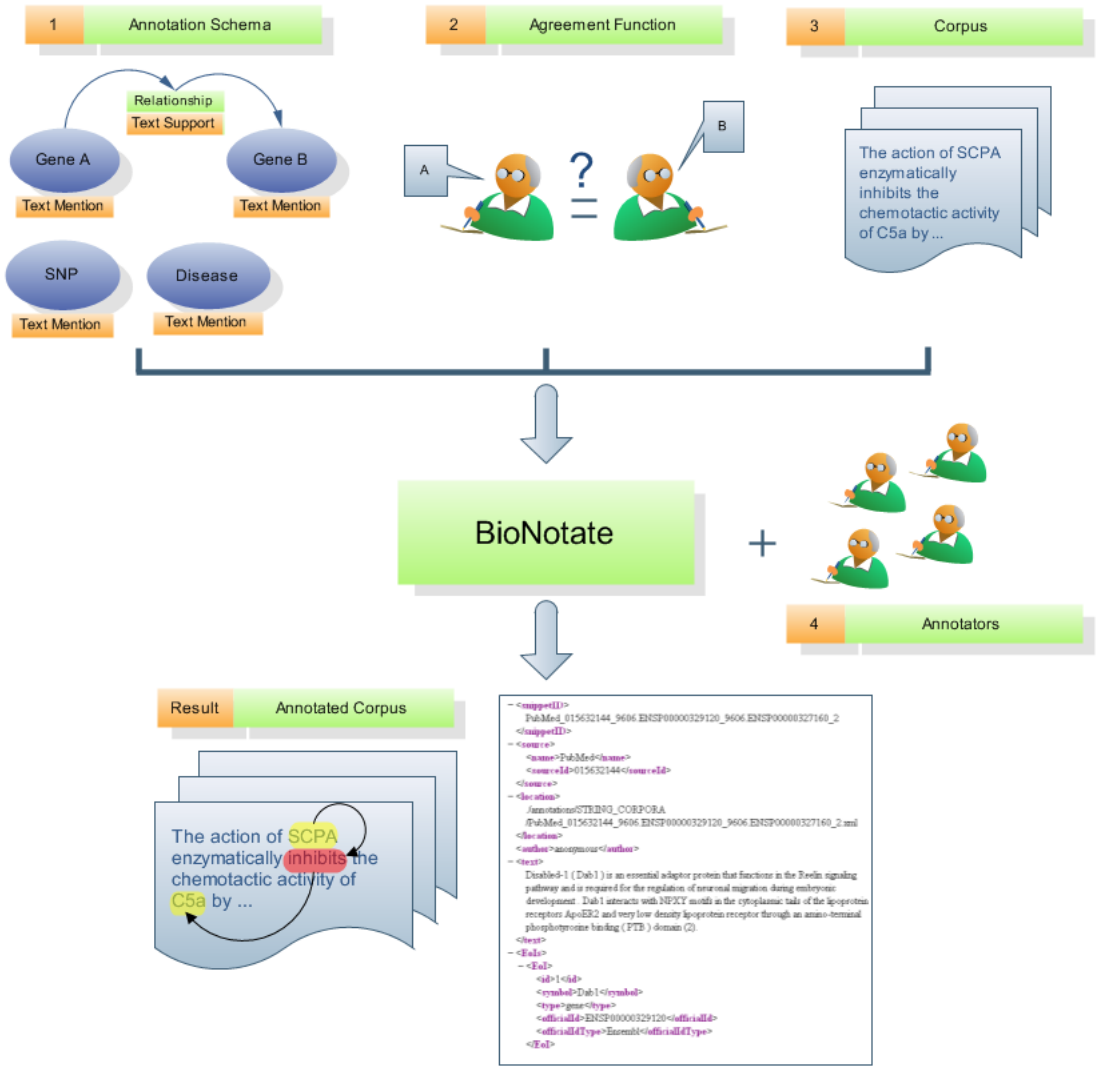


Figura 4.8: *Flujo de entrada y salida de las nuevas versiones de BioNotate. El usuario que administra el proceso de anotación proporciona el esquema de anotación, función de acuerdo entre anotadores y el corpus sin anotar.*

La figura 4.8 muestra el nuevo flujo de información de entrada y salida de BioNotate. El usuario que administra el proceso de anotación debe proporcionar el esquema de anotación, función de acuerdo y el corpus sin anotar a BioNotate. Tras la anotación distribuida del corpus utilizando BioNotate, el sistema proporciona las anotaciones de consenso (aquellas para las que hay acuerdo entre anotadores), según la función de acuerdo proporcionada.

El esquema de anotación debe ser definido en XML siguiendo un formato específico. Para definir el esquema de anotación, el administrador del sistema debe definir

el tipo de entidades que serán objeto de anotación (genes, proteínas, enfermedades, SNPs, etc.), las relaciones de interés entre estas entidades (por ejemplo, proteína-proteína, gen-enfermedad, etc.), y las preguntas que deben formularse al anotador durante el proceso de anotación (con las respuestas posibles que se ofrecen al anotador).

También se ofrece la posibilidad de que las entidades anotadas sean normalizadas por el anotador utilizando identificadores o términos de ontologías o recursos extendidos como Ensembl (`www.ensembl.org`), Uniprot (`www.uniprot.org`), MeSH (Medical Subject Headings, `http://www.nlm.nih.gov/mesh/`), Human Disease Ontology (DOID, `http://diseaseontology.sourceforge.net/`), etc. En este caso, el administrador debe seleccionar el recurso adecuado para cada entidad que se desea normalizar.

La función de acuerdo entre anotadores debe ser definida en función del esquema de anotación implementado y del nivel de consenso deseado para el corpus. A mayor nivel de consenso exigido, mayor número de anotaciones serán necesarias para alcanzar ese consenso, y mayor calidad tendrá el corpus resultante. La función de consenso debe definirse en Perl e integrarse en el código que se ejecuta del lado del servidor en BioNotate.

El desarrollo de esta extensión de BioNotate ha generado un gran interés en la comunidad científica, y algunos proyectos conjuntos con otras instituciones y empresas están surgiendo para anotar textos científicos de distintos ámbitos y siguiendo diferentes esquemas. El proyecto AutismNotate, que se describe en la sección siguiente, es una de estas colaboraciones.

4.6.2 AutismNotate.

La posibilidad de utilizar BioNotate para implementar distintos esquemas de anotación, tal y como se ha descrito en la sección 4.6.1, permite la utilización de este sistema en otros ámbitos distintos a aquellos para los que fue originariamente concebido. En esta sección, describimos brevemente la aplicación AutismNotate, disponible en `http://bionotate.hms.harvard.edu`, desarrollada en colaboración con la Universidad de Harvard y la empresa Alias-I Inc. (`http://alias-i.com/`) para la anotación de textos biomédicos relacionados con el Autismo.

Actualmente, existen más de 14,000 artículos relacionados con el Autismo (ASD) en la literatura publicada desde 1943. Los buscadores actuales no pueden diferenciar

cuáles de estos artículos mencionan un gen en un contexto relevante para la investigación genética. De este modo, estimamos que sólo 1 de cada 8 artículos devueltos por un buscador actual, trata realmente sobre la genética del ASD. El primer objetivo del proyecto AutismNotate es, utilizando la ayuda de anotadores humanos, crear un sistema que sea capaz de aprender la diferencia entre abstracts que hablan sobre genes y aquellos que no hablan sobre genes. Esto nos permitiría diseñar un sistema de búsqueda que detecte automáticamente artículos sobre ASD útiles para la investigación genética.

De este modo, el objetivo 1 del proyecto AutismNotate es la anotación distribuida por parte de anotadores humanos de textos sobre ASD que contienen menciones de genes. En particular, pedimos que el anotador centre su atención sobre un nombre de gen/proteína que ha sido previamente detectado mediante técnicas automáticas de NER y confirme, a través de la lectura del contexto en el que se encuentra, si dicha mención se corresponde, en efecto, con el nombre de un gen/proteína. Por ejemplo, los extractos siguientes contienen menciones de genes:

`In particular SERPINA3 showed increased mRNA levels in schizophrenia.`
`Notably, knockdown of this mutant (mt) p53 reduced cell viability and exerted antitumor activity equivalent to high doses of several chemotherapeutic agents`

Mientras que los siguientes no contienen nombres de genes:

`We soon found out that the P-51 Mustang was indeed a different breed of airplane.`
`All studies involved multi-item measures of attitude (Aact) and PBC items derived from pilot testing.`

La figura 4.9 muestra una captura de la página principal del proyecto. La figura 4.10 muestra la nueva interfaz de anotación. La adaptación de BioNotate para este proyecto ha sido inmediata gracias a las herramientas de flexibilización del esquema de anotación descritas en la Sección 4.6.1. En este caso, el esquema de anotación consta de un solo tipo de entidad (gen/proteína), y la siguiente pregunta es formulada al anotador:
Does the highlighted word/phrase refer to a gene or protein?
Presentando *Yes, No, Not sure* como posibles respuestas. La función de acuerdo en-

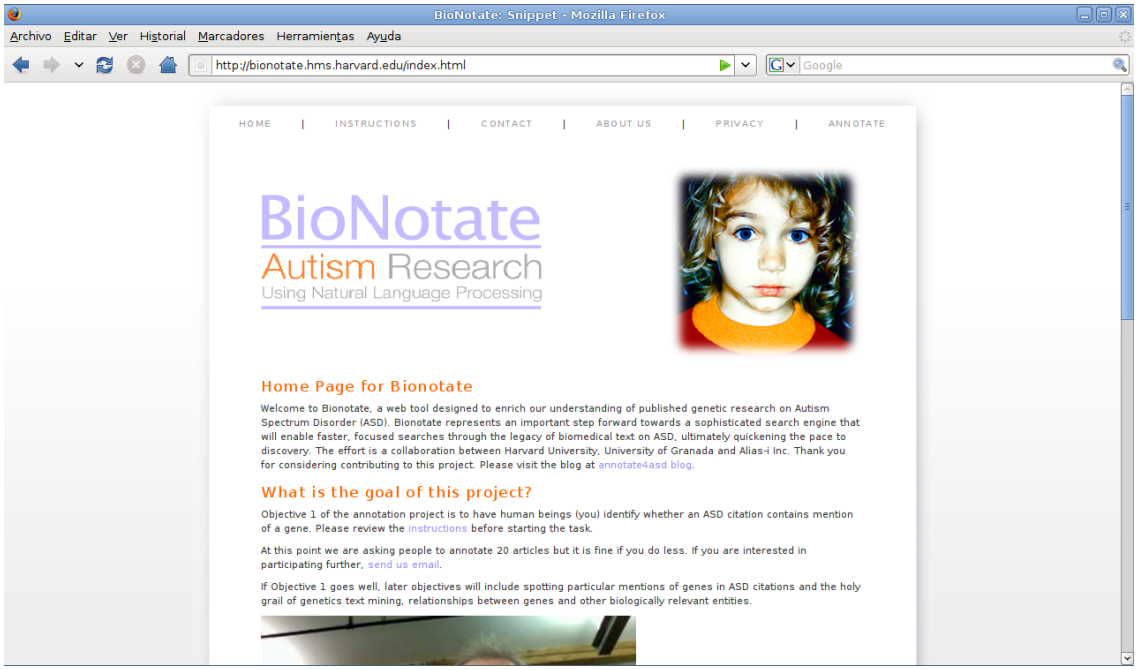


Figura 4.9: *Página principal del proyecto AutismNotate, que utiliza BioNotate para anotar textos sobre autismo. Disponible en: http://bionotate.hms.harvard.edu*

tre anotadores es simple: existirá acuerdo significativo entre dos anotaciones de dos usuarios distintos cuando la respuesta a la pregunta anterior sea la misma.

Para la detección automática de genes y proteínas en los textos, utilizamos la herramienta de NER de LingPipe [65]. De este modo, facilitamos la tarea del anotador, que simplemente debe validar si la mención marcada en el texto de forma automática se corresponde realmente con un gen o proteína. Este simple caso de uso muestra el enorme potencial de una herramienta que combine esfuerzos manuales y automáticos para la extracción de información de textos biomédicos, idea que es explorada en más detalle en la siguiente sección.

Próximos objetivos para el proyecto incluyen la participación de anotadores humanos para detectar menciones específicas de un gen en un abstract sobre ASD y la detección de relaciones entre estos genes en dichos abstracts.

Figura 4.10: *Interfaz de anotación de AutismNotate construida sobre BioNotate. Disponible en: http://bionotate.hms.harvard.edu/autism/snippet.html*

4.6.3 BioNotate y la Web Semántica.

Esta sección presenta una extensión de BioNotate que hace uso de la Web Semántica para la publicación y difusión de las anotaciones efectuadas. Este nuevo sistema se construye en base a la Web Social, para la anotación colaborativa y distribuida de textos biomédicos, y la Web Semántica, para la difusión de los resultados en formatos enriquecidos semánticamente, haciéndolos fácilmente accesibles. El propósito de esta extensión es crear un sistema en el que los usuarios pueden gestionar la literatura biomédica y el conocimiento extraído de la misma, compartir este conocimiento con la comunidad y descubrir, mediante este esfuerzo conjunto, relaciones que permanecían ocultas en la literatura.

Nuestra propuesta se basa en la combinación de herramientas automáticas de NER con la anotación manual y colaborativa de los textos y la normalización de entidades de interés, para una identificación más efectiva de hechos biológicos de interés en textos científicos. Además, el sistema permite que los hechos biológicos identificados se representen utilizando lenguajes estándar para que sea posible conectarlos con otros recursos de la Web Semántica.

El sistema propuesto proporciona servicios y herramientas web para:

1. La creación y anotación automática de un corpus de texto a partir de una consulta formulada por el usuario.
2. La corrección y normalización de estas anotaciones mediante la anotación manual del corpus utilizando la plataforma de anotación de BioNotate.
3. La publicación de las anotaciones finales en formato RDF.
4. El acceso a las anotaciones mediante un navegador para datos vinculados (*Linked Data*).

4.6.3.1 Arquitectura del Sistema.

El sistema propuesto está compuesto por cinco módulos básicos que cubren las distintas etapas del proceso de anotación: administración, búsqueda, anotación automática, anotación manual y publicación. La figura 4.11 muestra un esquema del sistema completo, en el que se interconectan estos cinco módulos.

Módulo de Administración.

El módulo de administración permite a los usuarios definir el problema, el esquema de anotación y el formato de los snippets que será utilizado en las tareas de anotación

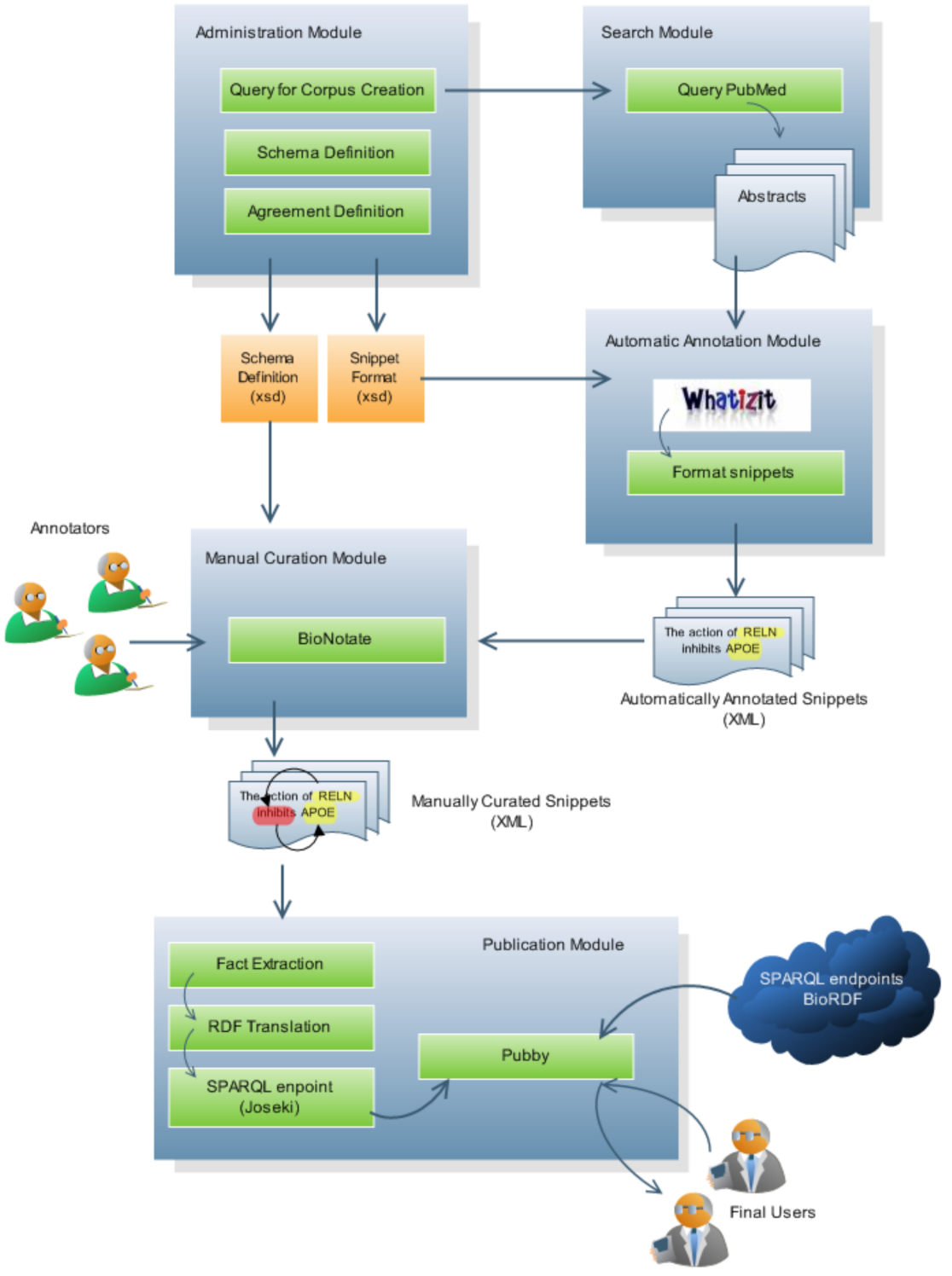


Figura 4.11: *Diagrama de funcionamiento de un sistema basado en BioNotate que combina la anotación automática y manual de textos para identificar hechos biológicos de interés en textos científicos. Los resultados se publican sacando partido de tecnologías de la Web Semántica para una mayor difusión e impacto en la comunidad científica.*

automática y manual. Este módulo está constituido por una interfaz de usuario intuitiva y sencilla, en la que el administrador puede definir las entidades y relaciones de interés para la tarea de anotación y proporcionar la función de acuerdo entre anotadores, todo ello utilizando la funcionalidad disponible en la extensión de BioNotate descrita en la Sección 4.6.1. Los administradores pueden cargar en la aplicación su propio corpus o bien crearlo utilizando las funciones de búsqueda y recuperación de artículos de PubMed descritos en el siguiente apartado.

Módulos de búsqueda y anotación automática.

En caso de que el administrador desee crear un corpus a partir de los abstracts obtenidos como respuesta a una consulta sobre Pubmed, el módulo de búsqueda le proporciona dicha funcionalidad. Éste módulo remite la consulta formulada por el administrador a PubMed y recupera los abstracts (y textos completos, si están disponibles) devueltos como resultado dicha consulta. Estos abstracts son presentados al administrador, que puede refinar la consulta o eliminar abstracts irrelevantes antes de salvar el resultado para generar el corpus.

El módulo de anotación automática permite utilizar sistemas automáticos de detección de entidades de interés (NER), o cualquier otro sistema existente de text-mining, para efectuar una primera anotación del texto que permita agilizar el proceso posterior de anotación manual. Nuestro sistema utiliza actualmente los servicios web de Whatizit [219, 260] para anotar las entidades de interés y relaciones entre elementos seleccionados por el usuario. Whatizit es un sistema de NER basado en tecnología Java para la búsqueda de correspondencias entre el texto y extensos recursos terminológicos, considerando variaciones morfológicas [180]. Whatizit también considera características sintácticas y etiquetas POS obtenidas con *TreeTagger* [276]. Existen distintos módulos disponibles en Whatizit para la identificación de NE o de relaciones entre entidades, entre los que destacamos los siguientes:

- *whatizitSwissprot*: centrado en la identificación de genes y proteínas y su normalización contra UniProt.
- *whatizitChemical*: centrado en la búsqueda de nombres químicos en base a la terminología de ChEBI [85] y al sistema de NER OSCAR3 [76].
- *whatizitDisease*: centrado en la identificación de nombres de patologías y enfermedades utilizando terminología extraida de MEDLINE.
- *whatizitDrugs*: para identificar fármacos utilizando la terminología DrugBank (`http://redpoll.pharmacy.ualberta.ca/drugbank/`).

- *whatizitGO*: para identificar términos de Gene Ontology.
- *whatizitOrganism*: identifica nombres de especies y organismos utilizando la taxonomía de NCBI.
- *whatizitProteinInteraction*: identifica relaciones proteína-proteína (gen-gen) utilizando Protein Corral (`www.ebi.ac.uk/Rebholz-srv/pcorral`).
- *whatizitSwissprotGo*: detecta relaciones entre proteínas(genes) y términos GO utilizando los léxicos de UniProtKb/Swiss-Prot.

Whatizit implementa estos y otros módulos y combinaciones de los mismos. La lista completa puede ser consultada en `http://www.ebi.ac.uk/webservices/whatizit/info.jsf`.

Módulo de anotación manual y normalización de entidades basado en BioNotate.

La anotación manual y colaborativa del corpus generado se basa en BioNotate. El administrador del sistema puede definir el esquema de anotación que desea implementar y la función de acuerdo entre anotadores a través del módulo de administración. Además, se permite que el administrador proporcione enlaces a bio-ontologías u otros recursos terminológicos para la normalización de las entidades encontradas en el texto. De este modo, cuando el anotador marca una entidad y le asigna una etiqueta, también debe proporcionar al sistema su identificador oficial en la ontología o base de datos correspondiente.

El entorno de anotación de BioNotate ha sido modificado para recoger la posibilidad de que el anotador normalice las entidades conforme a recursos externos. Este entorno modificado se muestra en la Figura 4.12.

Como caso de uso, el sistema desarrollado por nuestro equipo, que está actualmente disponible en la web `http://genome2.ugr.es/bionotate`, implementa un esquema de anotación de relaciones proteína-proteína (gen-gene), en el que las proteínas (genes) se normalizan contra UniProt y las palabras clave de la interacción se normalizan contra la ontología PSI-Mollecular Interactions (PSI-MI).

De este modo, el nuevo proceso de anotación se describe a continuación:

1 Indicar Si/No en función de si el texto implica que existe una relación directa entre las proteínas de interés.

2 Marcar la secuencia de palabras mínima y más significativa que justifique la respuesta anterior (si existe). El texto marcado se etiquetará como INTERACTION.

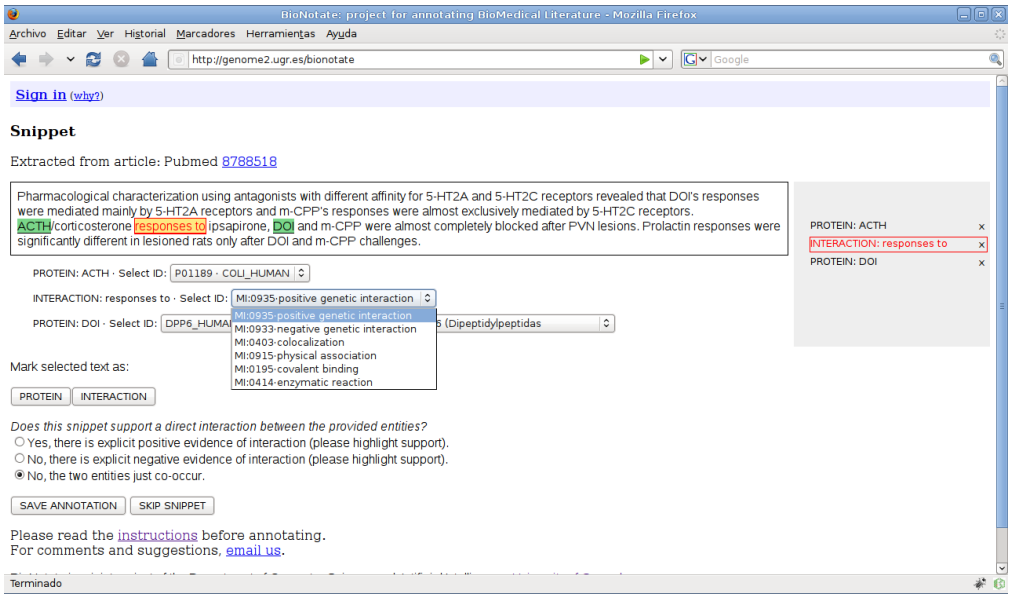


Figura 4.12: *Captura de pantalla del sistema de anotación BioNotate que muestra la posibilidad de normalizar las entidades marcadas contra ontologías y recursos terminológicos. La figura muestra dos proteínas normalizadas con UniProt y una interacción normalizada con la ontología PSI-Mollecular Interactions.*

3 Localizar y marcar en el texto la mención de cada una de las entidades de interés que es esencial para expresar la relación entre las mismas (en su caso). Definimos estas menciones como aquellas que, si fueran omitidas o reemplazadas por menciones de otras entidades, modificarían el mensaje transmitido en el texto de forma que no se expresaría la misma relación.

4 Proporcionar un término de la ontología PSI-MI para el texto etiquetado como INTERACTION en [2] e identificadores de UniProt para las dos proteínas etiquetadas en [3].

La función de acuerdo entre anotadores se define como sigue. Se dice que k anotaciones cumplen un grado de acuerdo mínimo cuando se satisfacen las siguientes condiciones:

1 La respuesta Si/No es la misma.

2 La secuencia de tokens marcada con las etiquetas PROTEIN se solapan completamente y los identificadores UniProt proporcionados para las dos entidades son los mismos.

3 La secuencia de tokens marcada con la etiqueta INTERACTION se solapan significativamente, permitiendo hasta 1 token diferente respecto a la secuencia más

corta para cada una de las parejas de anotaciones de tipo INTERACTION de las *k* anotaciones. Además, el identificador PSI-MI proporcionado para la interacción debe ser el mismo.

Módulo de publicación de anotaciones

Las tecnologías basadas en la Web Semántica permiten la interconexión de datos de distintas fuentes mediante el establecimiento de conexiones o vínculos entre los mismos (*Linked Data*). Estos vínculos explícitos permiten a los investigadores navegar de forma transparente entre datos de distintos recursos y descubrir relaciones que desconocían. La utilización de una representación estándar para la información (RDF) y de mecanismos estándares de accesso a la misma (HTTP), permite a los navegadores de Web Semántica y a los motores de búsqueda especializados acceder y procesar este tipo de información [158].

De este modo, proponemos que el resultado del proceso de anotación descrito en esta sección esté disponible a la comunidad en forma de *Linked Data*, y pueda ser conectado a otros recursos de la web como Bio2RDF or LODD. En particular, hemos dotado al sistema con un módulo de publicación, que aporta la funcionalidad necesaria para convertir las anotaciones a formato RDF (utilizando el lenguaje Notation3) o publicarlas como un punto de acceso SPARQL. SPARQL es un lenguaje de consulta capaz de recuperar y manipular información almacenada en formato RDF. Este módulo de publicación está implementado en base a dos proyectos *open-source* en Java: Joseki y Pubby.

Joseki es un motor HTTP que soporta el protocolo SPARQL y el lenguaje de consulta SPARQL RDF. Forma parte de Jena, un entorno Java para desarrollar aplicaciones de Web Semántica desarrollado en los laboratorios HP. Joseki está disponible en `http://www.joseki.org`.

Pubby es un sistema desarrollado por la *Free University of Berlin* que permite convertir un punto de acceso SPARQL en un servidor de *Linked Data*. Se implementa como una aplicación web Java y está disponible en: `http://www4.wiwiss.fu-berlin.de/pubby/`.

4.7 Conclusiones.

En este capítulo se ha descrito BioNotate (`http://bionotate.sourceforge.net/`), una herramienta web de código libre para la anotación colaborativa de textos a través de una interfaz simple e intuitiva y utilizando un sistema estándar cliente-servidor. Esta herramienta proporciona los medios necesarios para crear un corpus de tamaño sustancial que permita la evaluación y desarrollo de métodos de detección de entidades biomédicas y las relaciones entre ellas en textos biomédicos. Además, se ha presentado un estudio que revisa los distintos corpora de textos biomédicos disponibles centrados en la anotación de genes/proteínas, relaciones entre ellos y anotaciones sintácticas, justificando la necesidad de nuevos esfuerzos de anotación comunitarios para la anotación de corpora siguiendo esquemas adecuados. Esperamos que la herramienta presentada en este capítulo suponga un beneficio mutuo para la biología (que se beneficia de la extracción de conocimiento de la literatura biomédica) y las técnicas de NLP (beneficiadas de la disponibilidad de corpora para desarrollar y mejorar las herramientas de text-mining), además de facilitar las tareas de anotación manual, haciendo que los anotadores corrijan las anotaciones efectuadas de forma automática por herramientas de text-mining. En la actualidad, BioNotate incorpora los servicios de Whatizit para la identificación automática de entidades biomédicas de interés, aunque ya se estudia su ampliación a otras técnicas de NER e identificación de relaciones en los textos.

Además describimos un método para la creación de un corpus piloto centrado en la extracción de relaciones entre genes asociados con autismo para la validación de la red genética del proyecto Autworks. BioNotate puede ser cómodamente alimentado con nuevos conjuntos de snippets, facilitando el desarrollo de esfuerzos similares al llevado a cabo para la anotación de corpora para autismo u otras enfermedades. Además, puesto que la herramienta es de libre distribución, puede ser descargada y utilizada en cualquier lugar, permitiendo el desarrollo de múltiples pequeños esfuerzos de anotación paralelos, cuyo resultado pueda ser integrado en un único recurso uniforme. La consistencia del corpus resultante está garantizada siempre que los esfuerzos paralelos de anotación implementen el mismo esquema. Por ejemplo, para la anotación de relaciones gen-gen y gen-enfermedad descrito en la sección 4.4, se hace indispensable que, para cada relación anotada, se indiquen las ocurrencias exactas de las entidades que interaccionan y el texto que soporta explícitamente la existencia de dicha relación.

Las extensiones para BioNotate descritas en este capítulo van dirigidas a aumen-

tar la difusión y uso de la herramienta en la comunidad científica. La extensión que permite que BioNotate implemente cualquier esquema de anotación definido por el usuario, nos ha permitido integrar esta herramienta en entornos de anotación colaborativos como AutismNotate. Además, esperamos que la posibilidad de publicar y difundir los hechos biológicos anotados gracias a la extensión descrita en la seccion 4.6.3, promueva el uso de la herramienta en la comunidad y facilite su integración con otros recursos *online* .

El sistema propuesto puede ser mejorado como resultado de una re-alimentación entre los procesos de anotación manual, el entrenamiento de sistemas de text-mining y el uso de herramientas automáticas para facilitar la anotación manual. Un aspecto a mejorar que encontramos durante la anotación del corpus piloto sobre autismo (sección 4.5) es la constitución poco balanceada del corpus considerado, es decir, la existencia de una gran mayoría de snippets que no soportan una relación entre genes. Actualmente, estudiamos la utilización de herramientas más sofisticadas para seleccionar automáticamente los snippets e identificar candidatos a menciones de genes para crear un corpus mejor balanceado. También estamos desarrollando métodos para clasificar automáticamente los snippets utilizando heurísticas poco complicadas para identificar snippets candidatos a constatar una relación.

Capítulo

5

Identificación automática de diagnósticos en historias clínicas. Caso de estudio en obesidad

5.1 Motivación y objetivos.

El entendimiento automático de los historiales clínicos supone un reto importante dentro del campo de la minería de textos en medicina. Los historiales clínicos constituyen una rica y extensa fuente de información para la investigación médica [252]. Este conocimiento se almacena principalmente en forma de texto libre, y contiene abundante información médica relacionada con síntomas, diagnósticos, tratamientos e historias clínicas. Sin embargo, los textos de los historiales clínicos son inherentemente desestructurados, redundantes, y contienen gran cantidad de acronismos, abreviaciones y lenguaje altamente especializado, lo que hace que el análisis automático de este tipo de textos se convierta en una tarea altamente costosa y difícil. Además, la normativa sobre privacidad y protección de datos de carácter personal hace que la publicación de este tipo de corpora para la investigación en este campo sea muy escasa [317].

Existen numerosos trabajos que proponen el uso de ontologías, léxicos y heurísticas especializadas para, a partir del texto de historias clínicas, diagnosticar de forma automática un trastorno concreto, utilizando conocimiento experto en ese ámbito (ver Sección 1.3.2). Sin embargo, dado que estas técnicas se diseñan *ad-hoc* para una enfermedad concreta, su aplicación a otras enfermedades o diagnósticos no resulta efectiva.

En este capítulo se presenta un clasificador basado en Regresión Logística para la identificación del trastorno principal y comorbilidades de un paciente a partir del análisis del texto de su historial clínico. Este trabajo está basado en la rápida adaptación de Ling-Pipe [65], una popular plataforma de software para text-mining. En particular, proponemos distintos clasificadores de Regresión Logística que utilizan características basadas en las palabras de los historiales clínicos para asignar a cada historial su diagnóstico correcto. El sistema que proponemos no utiliza ontologías ni heurísticas atadas a un dominio concreto, por lo que puede ser rápidamente adaptado a cualquier ámbito.

Como caso de estudio, este capítulo muestra los resultados del clasificador propuesto para la identificación de obesidad y quince de sus comorbilidades más relacionadas [320] a partir del texto de los historiales clínicos de los pacientes. En particular, aplicamos los clasificadores propuestos sobre un corpus que consta de varios cientos de historiales clínicos proporcionados por *Partners Healthcare* (`http://www.partners.org/`) para los cuales el objetivo es identificar el diagnóstico principal y comorbilidades en pacientes que fueron evaluados de obesidad o diabetes. Además, implementamos otros clasificadores para efectuar una comparativa con los sistemas propuestos para mostrar el potencial y la adecuación de las técnicas propuestas para el análisis de historiales clínicos.

Este capítulo se estructura como sigue. En primer lugar, la sección 5.2 presenta una breve revisión a las técnicas de text-mining aplicadas sobre historiales clínicos. La sección 5.3 describe el corpus utilizado para el caso de estudio sobre obesidad y otras patologías relacionadas. En la sección 5.4 se presentan los fundamentos del clasificador basado en regresión logística que proponemos, junto con las descripciones de distintos conjuntos de características empleados y la arquitectura del sistema completo. En 5.5 se presentan los resultados obtenidos para el caso de estudio en obesidad, justificando las mejoras introducidas por las distintas estrategias empleadas en los clasificadores (5.5.3). Finalmente, la sección 5.6 presenta las conclusiones de este estudio.

5.2 Trabajo previo en identificación automática de diagnosis en historias clínicas.

Recientemente, han surgido numerosos métodos para mapear de forma automática una pieza de texto de un historial clínico en conceptos estandarizados de ontologías médicas como UMLS (*Unified Medical Language System*) [110, 204, 294, 87]. Sin embargo, la identificación de conceptos estandarizados en el texto de una historia clínica no garantiza la asignación del diagnóstico correcto. El texto clínico (ver Tabla 1.6) puede contener conceptos relevantes que están negados (por ejemplo, "No murmurs, rubs or gallops", "Denies alcohol use"), que hacen referencia a miembros familiares en lugar de al propio paciente ("The patient's mother had a myocardial infarction"), que se refieren al pasado ("He had no history of previous hypertension") o que son inciertos ("questionable history of gout", "We recommend polysomnography evaluation for OSA"). Por tanto, una clasificación fiable de textos clínicos para asignar un diagnóstico correcto requiere tener en cuenta el contexto, la co-ocurrencia de varias palabras en el texto, en lugar de considerar conceptos aislados [342].

El trabajo previo en clasificación de historias clínicas para la asignación de diagnóstico [239, 197] está basado en el uso de ontologías asociadas a un dominio específico, léxicos especializados y expresiones regulares definidas manualmente para un caso de estudio concreto. La creación y mantenimiento de todos estos recursos requiere de un amplio conocimiento experto, y su utilización hace que el sistema esté fuertemente ligado a un dominio (enfermedad) específico. De este modo, el rendimiento de estos sistemas es muy pobre cuando se utilizan para el diagnóstico de otros trastornos distintos de aquellos para los que fueron diseñados. Además, el hecho de que estos sistemas no estén disponibles, hace que la comparación de las nuevas propuestas con enfoques ya existentes se mantenga como uno de los retos en este campo [247, 319, 320].

A diferencia de este tipo de métodos, nuestra propuesta muestra una metodología orientada al diseño rápido de un clasificador para cualquier dominio, utilizando herramientas disponibles de text-mining. Esta metodología se ha mostrado muy exitosa en campos relacionados como la ocultación de datos personales en historias clínicas [328]. En nuestra propuesta adaptamos LingPipe [65], una herramienta de Alias-I, para acometer la asignación de diagnósticos a historiales clínicos reales por medio de un clasificador basado en regresión logística. En particular, nos centraremos en un caso de estudio para Obesidad y algunas de sus comorbilidades más relacionadas [320].

5.3 Descripción del corpus de historiales clínicos.

El corpus consta de los textos completos de 1237 historias clínicas proporcionadas por el consorcio *Partners Healthcare*. Todos los datos personales de estos historiales han sido sustituidos por nombres, fechas y lugares falsos para adecuarse a la normativa de privacidad y protección de datos de carácter personal de la Oficina de Privacidad de Información (HIPAA) del Departamento de Salud y Servicios Humanos del gobierno de EEUU. Estas historias clínicas pertenecen a pacientes que fueron evaluados de obesidad o diabetes. Este corpus se centra, en particular, en la obesidad y quince de sus comorbilidades más frecuentes según la Base de Datos de Investigación de *Partners Healthcare*: *Diabetes mellitus*, Hipercolesterolemia (*Hypercholesterolemia*), Hipertriglicemia (*Hypertriglyceridemia*), Hipertensión (*Hipertension*), Cardiopatía isquémica (*Atherosclerotic CV disease*, CAD), Fallo cardiaco (*Heart Failure*, CHF), Enfermedad vascular periférica (*Peripheral vascular disease*, PVD), Insuficiencia venosa (*Venous insufficiency*), Osteoartritis (*Osteoarthritis*, OA), Apnea del sueño (*Obstructive sleep apnea*, OSA), Asma (*Asthma*), Reflujo gastroesofágico (*Gastroesophageal reflux disease*, GERD), cálculos biliares/colecistectomía (*Gallstones/Cholecystectomy*), Depresión (*Depression*) y Gota (*Gout*). El corpus fue anotado por dos expertos en obesidad del Centro de Sobrepeso del Hospital General de Massachussets (Massachussets General Hospital -MGH-). Para cada enfermedad objeto de estudio, los expertos evaluaron si el paciente padece la enfermedad ("Y"), no la padece ("N"), es cuestionable si la padece o no ("Q") o la enfermedad no se menciona en el historial ("U"). Para cada paciente los expertos proporcionaron:

- Evaluaciones *textuales*: evaluaciones basadas exclusivamente en lo que el texto constata explícitamente.
- Evaluaciones *intuitivas*: evaluaciones basadas en lo que el texto implica, aunque no se mencione explícitamente.

En el caso de las evaluaciones intuitivas, sólo se utilizan las etiquetas "Y", "N" y "Q" (en este caso no tiene sentido la etiqueta "U" porque la evaluación no se basa en menciones explícitas únicamente). El voto de los especialistas determinó la evaluación final para cada paciente y enfermedad. En caso de desacuerdo, un tercer especialista resolvió el empate.

El conjunto de entrenamiento incluye el 60 % de los historiales con evaluación textual para obesidad de "Y", "N", "Q" y "U" y sus correspondientes evaluaciones intuitivas. El conjunto de test lo componen el resto de los historiales. La distribución

de historiales en las distintas clases no es uniforme: la media por enfermedad para las evaluaciones intuitivas es 31.2 % para la clase 'Y', 68.56 % para la clase 'N' y 0.24 % para la clase 'Q'. Para las evaluaciones textuales los porcentajes son: 'Y':27.6 %, 'N':0.75 %, 'Q':0.33 %, 'U':71.3 %.

Las historias clínicas de este corpus típicamente incluyen información textual acerca de, entre otros, el diagnóstico principal, diagnósticos secundarios, histórico de la enfermedad actual, medicación antes de admisión, historial médico, historial familiar, historial social, alergias, exámen físico en el momento de admisión, complicaciones, medicación tras el alta, etc. Toda esta información está proporcionada como texto libre en lenguaje natural y sin estructura. La Tabla 1.6 muestra un extracto de un historial a modo de ejemplo.

5.4 Clasificación basada en Regresión Logística para Historias Clínicas.

El sistema de clasificación que proponemos está basado en modelos de Regresión Logística Multinomial.

5.4.1 Regresión Logística.

Un modelo de Regresión Logística Multinomial clasifica vectores reales x de dimensión d, $x \in \mathbb{R}^d$, en una de las k clases posibles $c \in \{0, \ldots, k-1\}$ usando $k-1$ parámetros vectoriales (coeficientes de regresión), uno para cada clase, $\beta_0, \ldots, \beta_{k-2} \in \mathbb{R}^d$:

$$p(c \mid x, \beta) = \begin{cases} \frac{\exp(\beta_c \cdot x)}{Z_x} & \text{si } c < k-1 \\ \frac{1}{Z_x} & \text{si } c = k-1 \end{cases} \tag{5.1}$$

donde $\beta_c \in \mathbb{R}^d$ denota los coeficientes de regresión para la clase c y $\cdot$ denota el producto escalar:

$$\beta_c \cdot x = \sum_{i<d} \beta_{c,i} \cdot x_i \tag{5.2}$$

Y el factor de normalización Z_x se define como:

$$Z_x = 1 + \sum_{c<k-1} \exp(\beta_c \cdot x) \tag{5.3}$$

Para estimar los $k-1$ coeficientes de regresión, notados como $beta{:}\beta_0,\ldots,\beta_{k-2}$, que necesita el modelo de regresión logística, utilizamos un decisor máximo a posteriori (*maximum a posteriori*, MAP) que minimiza la función de error que definimos a continuación. Para encontrar este mínimo utilizamos un método de optimización basado en Descenso Estocástico de Gradiente.

Una vez estimados los coeficientes β, calculamos para cada elemento del conjunto de test x_i, su valor de probabilidad $p(c_j \mid x_i, \beta_{c_j})$ para cada clase c_j. La clase que presente mayor valor de probabilidad será asignada a x_i.

Suponemos que las probabilidades *a priori* siguen distribuciones de Laplace de media 0 y varianza σ^2: $Laplace(0,\sigma^2)$, con σ^2 variando para cada clasificador. El coeficiente β_0 es la ordenada en el origen, y asumiremos para ella una probabilidad uniforme (que es equivalente a asumir distribución $Laplace(0,\infty)$). Para el resto de coeficientes $\beta_1,\ldots,\beta_{k-2}$ se asume la misma varianza σ^2 y, por tanto, la misma distribución de probabilidad $Laplace(0,\sigma^2)$. Asumimos que los coeficientes se distribuyen según una Laplaciana en lugar de otras distribuciones como la normal o Gaussiana porque trabajos previos en clasificación de textos han mostrado que la Laplaciana es mucho más robusta y proporciona mejores resultados tras validación cruzada que otras distribuciones [117, 126].

Dado un conjunto de n datos de entrenamiento $D = \langle x_j, c_j\rangle_{j<n}$, con $x_j \in \mathbb{R}^d$ y $c_j \in \{0,\ldots,k-1\}$, la verosimilitud logarítmica de estos datos en base a un modelo con parámetros β es:

$$\log p(D \mid \beta) = \log \prod_{j<n} p(c_j \mid x_j, \beta) = \sum_{j<n} \log p(c_j \mid x_j, \beta) \tag{5.4}$$

Suponiendo que los coeficientes β se distribuyen según $Laplace(0,\sigma_i^2)$ para cada dimension $i<d$, podemos expresar la probabilidad a priori para la matriz β como:

$$p(\beta \mid \sigma^2) = \prod_{c<k-1} \prod_{i<d} \mathsf{Laplace}(0,\sigma_i^2)(\beta_{c,i}) \tag{5.5}$$

donde:

$$\mathsf{Laplace}(0,\sigma_i^2)(\beta_{c,i}) = \frac{\sqrt{2}}{2\sigma_i} \exp\left(-\sqrt{2}\,\frac{\mid \beta_{c,i} \mid}{\sigma_i}\right) \tag{5.6}$$

La función de error dependiente de los parámetros β, los datos de entrenamiento D y la varianza para la distribución a priori σ^2, es la suma de la verosimilitud y la

probabilidad a priori:

$$\begin{aligned} \mathrm{Err}(\beta, D, \sigma^2) &= -\log p(D \mid \beta) - \log p(\beta \mid \sigma^2) \\ &= \sum_{j<n} -\log p(c_j \mid x_j, \beta) + \sum_{0<i<d} -\log p(\beta_i \mid \sigma_i^2) \end{aligned}$$

Esta función es cóncava y por tanto tiene un mínimo único, que es el valor que maximiza la probabilidad a priori (MAP). Calculamos este mínimo mediante un método de optimización denominado Descenso de Gradiente Estocástico (*stochastic gradient descent*, SGD), basado en las derivadas parciales de la función de error [267]. El algoritmo básico para SGD es:

$\beta = 0$
for $epoch = 0$ to $maxEpochs$ **do**
  for trainingCase $(x, c\prime)$ in D **do**
    for category $c < numOutcomes - 1$ **do**
      $\beta_c -= learningRate(epoch) * grad(Err(x, c, c', \beta, \sigma^2))$
    end for
  end for
  if $epoch > minEpochsANDconverged$ **then**
    return β
  end if
end for

donde los parámetros $learningRate$, $minEpochs$, $maxEpochs$ y el criterio de convergencia deben ser proporcionados por el usuario.

Más detalles sobre el algoritmo SGD y el modelo de regresión logística multinomial implementado pueden consultarse en [64] y en el sitio `http://alias-i.com/lingpipe/demos/tutorial/logistic-regression/`.

5.4.2 Conjuntos de características.

El texto de los historiales clínicos necesita ser representado como un vector de características para ser pasado como entrada al clasificador propuesto [107]. Para ello se proponen tres representaciones posibles:

- N-gramas de caracteres. Nuestro primer enfoque utiliza n-gramas de caracteres como características, es decir, calculamos todas las secuencias de n caracteres que aparecen en el texto y contamos su número de ocurrencias en cada historial clínico. En particular, proponemos $n = 5$ para esta tarea, ya que nos ha

proporcionado mejores resultados experimentalmente. Asimismo, descartamos los 5-gramas que ocurran menos de 20 veces en el texto de los historiales del conjunto de entrenamiento. De este modo, reducimos el conjunto de características a unos 20,000 5-gramas distintos para cada tarea. Para formar los 5-gramas tenemos en cuenta los espacios en blanco y no llevamos a cabo ninguna normalización de caracteres, por ejemplo, los n-gramas derivados de la cadena "Anemia and GI" son: "Anemi", "nemia", "emia ", "mia a", "ia an", "a and", " and ", "and G" and "nd GI".

- Léxico enriquecido. El segundo conjunto de características propuesto consiste en representar cada texto como el conjunto de palabras que aparecen en el mismo y su número de ocurrencias. Por ejemplo, para la cadena "Anemia and GI" la representación sería: "Anemia": 1; "and":1; "GI":1. Además normalizamos las ocurrencias de los nombres de enfermedades mediante resolución de acrónimos y detección de sinónimos. Por ejemplo, todas las ocurrencias de las cadenas de texto: "Peripheral vascular disease", "Peripheral Artery Disease", "Peripheral Arterial Disease" y "PAD" o cualquiera de sus variantes por capitalización, que hacen referencia todas ellas a la misma patología, son reemplazadas por una cadena única, en este caso "PVD". Finalmente, también se incluyen características adicionales que consideramos informativas para la tarea de clasificación que llevamos a cabo:

  1. Reconocimiento de Medicación. Construimos diccionarios de nombres de medicamentos y las enfermedades para las que se suelen recetar a partir de los recursos online `www.rxlist.com` y `www.druglib.com`. De este modo, para cada nombre de medicamento detectado en el texto de un historial, añadimos la característica DISEASE_DRUG en el texto, donde DISEASE se corresponde con el nombre normalizado de la enfermedad o enfermedades para la que ese medicamento está prescrito, según el diccionario de asociaciones descrito anteriormente. Por ejemplo, el extracto "lantus 25 units subcu q.a.m." se representaría con las características:
     "lantus": 1; "25":1; "units":1; "subcu":1; "q.a.m": 1; "Diabetes_DRUG": 1
     ya que "lantus" es el nombre de un medicamento prescito para Diabetes.
  2. Tokens negados. Para aumentar la precisión en la clasificación de ejemplos negativos (clase "N"), se añaden características de la forma "NO_token" que resultan de la negación de tokens que siguen a una ocurrencia de las palabras "no", "not" o "without". Por ejemplo, para el extracto: "He had no

history of diabetes or hypertension", además de las características obtenidas de las palabras citadas en el mismo, se consideran las características: "NO_history" "NO_of" "NO_diabetes" "NO_or" "NO_hypertension".

- Léxico más relevante. La tercera representación propuesta se basa en la representación anterior. La novedad reside en que en este caso se seleccionan sólo aquellas características o tokens más informativos en términos de *Information Gain* [107]. En particular, construimos distintos léxicos utilizando los N tokens de mayor *Information Gain* para
  $N \in \{1, 2, 3, 4, 5, 10, 15, 20, 25, 30, 40, 50, 70, 100, 200\}$
  considerando todas las categorías y enfermedades. Este proceso ha sido llevado a cabo por separado para la evaluación textual e intuitiva, y el léxico reducido que produjo los mejores resultados en promedio fue seleccionado (los resultados detallados se muestran en la Sección 5.5.3).

5.4.3 Arquitectura del sistema.

El sistema de clasificación de historiales clínicos que proponemos se basa en la plataforma de desarrollo de text-mining llamada *LingPipe*. LingPipe es una *suite* de librerías de Java para Procesamiento de Lenguaje Natural (NLP) y *text-mining*. LingPipe implementa módulos de NLP de tokenización, *part-of-speech (POS) tagging*, reconocimiento de entidades (*named entity recognition*, NER) y clasificación de textos, entre otras.

Para implementar el clasificador de textos utilizamos el modelo de regresión logística implementado en LingPipe. La Figura 5.1 muestra la arquitectura del sistema durante el entrenamiento.

La entrada del clasificador es un vector de características que representa los historiales clínicos. Para obtener estas características a partir del texto de los historiales, tal y como se describe en la Sección 5.4.2, implementamos los módulos "Pre-processor", "Tokenizer" y "Feature Extractor" de la Figura 5.1. El módulo "Pre-processor" preprocesa los historiales clínicos, normalizando las menciones a enfermedades y nombres comerciales de medicamentos y procesando las negaciones según lo descrito en la Sección 5.4.2. El módulo "Tokenizer" divide el texto en n-gramas o palabras, según la representación de las características, descritas en la Sección 5.4.2, utilizada por el clasificador. El módulo "Feature Extractor" genera el vector de características a partir del texto mediante el recuento de las ocurrrencias de las distintas palabras o n-gramas en los historiales. Finalmente, se añade una característica única con valor 1.0 a cada

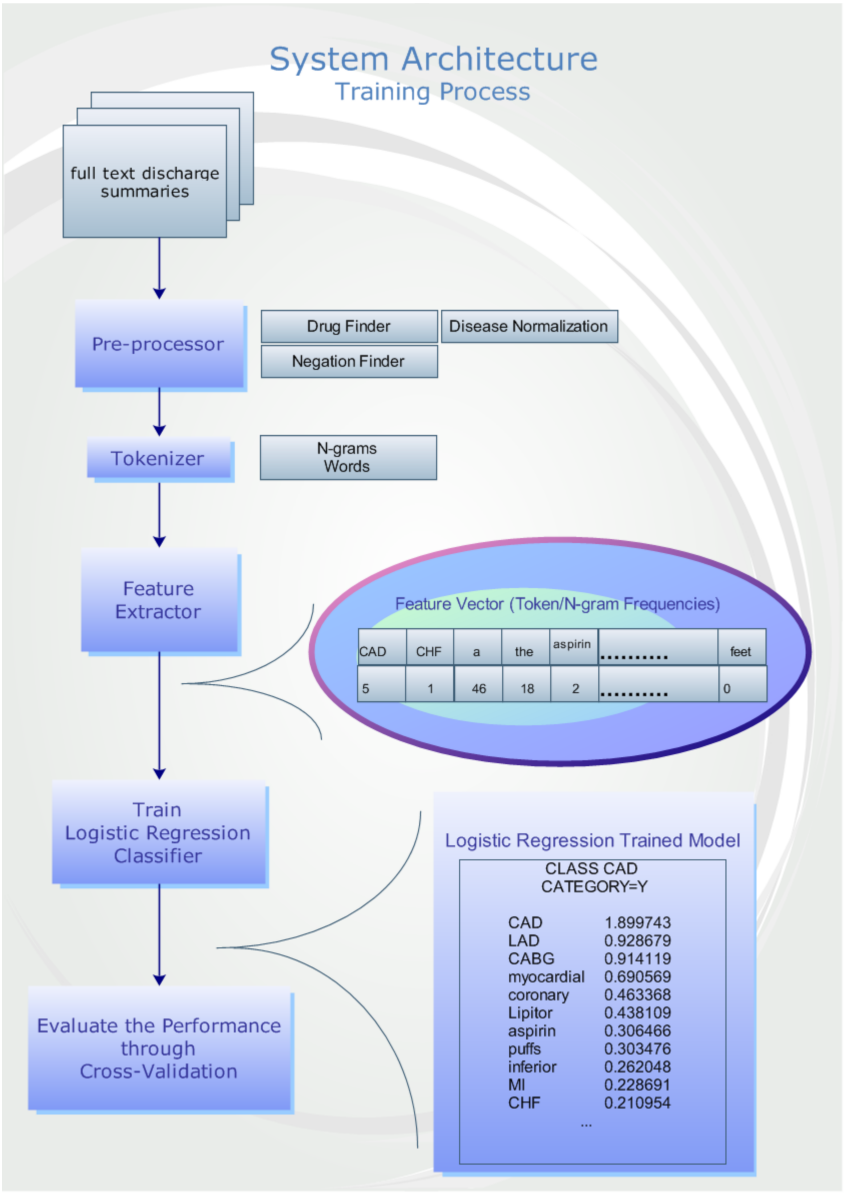


Figura 5.1: *Arquitectura del sistema clasificador de historiales clínicos.*

vector de características antes de realizar el entrenamiento y test para determinar el coeficiente β_0 según [117].

Una vez que los historiales clínicos se transforman en un conjunto de características numéricas, construimos el clasificador según el modelo de regresión logística descrito en la Sección 5.4.1. Para entrenar el clasificador implementamos un método de Enfriamiento Simulado (Annealing Schedule) [179] que calcula la tasa de aprendizaje (*learning rate*) para cada iteración del proceso de optimización SGD descrito en la sección 5.4.1. La elección de los parámetros de aprendizaje del modelo de re-

gresión logística se realiza mediante validación cruzada dividiendo el conjunto de entrenamiento en 4 subconjuntos. La evaluación de las distintas configuraciones se realiza en base a *precision*, *recall* y *f-measure*.

Los mejores resultados fueron obtenidos considerando distribuciones $Laplace(0,0{,}1)$ y Enfriamiento Simulado con tasa de enfriamiento exponencial, con tasa de aprendizaje inicial de 0,001 y ratio exponencial de decrecimiento de 0,9999.

Los modelos de regresión logística ya entrenados se representan como una lista de características y sus coeficientes asociados para cada clase (parámetros β). El análisis de estos modelos arroja algunas conclusiones de interés. Por ejemplo, no sorprende el hecho de que referencias a las enfermedades y otras palabras clave aparezcan asociadas a los coeficientes más altos y más bajos para la clasificación de las clases "Y" y "N"/"U" , respectivamente. Por ejemplo, "Venous Insufficiency", "Stasis", "Venous", "Cellulitis", "legs", "stockings" y "erythema" son algunas de las características con los coeficientes β más altos para la clase "Y" para la enfermedad *Venous Insufficiency*. Por tanto, la presencia de estas palabras en un historial médico apoya la clasificación del paciente dentro de la categoría "Y" para *Venous Insufficiency*. La Tabla 5.1 muestra otro ejemplo de un extracto de la lista de características más relevantes para la categoría "Y" de la enfermedad *Atherosclerotic CV disease (CAD)*. Podemos comprobar de forma intuitiva la fuerte relación entre la ocurrencia de estos términos en el texto de un historial clínico y el diagnóstico asociado.

5.5 Experimentación y Resultados.

5.5.1 Evaluación comparativa de la Regresión Logística para clasificación de textos.

Para evaluar el potencial de la regresión logística en comparación con otros clasificadores, implementamos un clasificador Naïve Bayes (NB) (descrito en la sección 1.2.4) y un clasificador (que denominamos *PickMostLikelyCategory* o PMLC) que asigna a cada historial el diagnóstico más probable, es decir el diagnóstico con más historiales asociados en el conjunto de entrenamiento. La Tabla 5.2 muestra un análisis comparativo de la bondad de clasificación utilizando estos tres métodos y considerando una representación de características basada en el léxico estándar que se deriva del texto de los historiales.

Enfermedad	CAD
Tipo evaluación	intuitiva
#Características	331
#Categorias	3
Categoría	Y
INTERCEPT	2.688587
CAD	1.899743
LAD	0.928679
CABG	0.914119
Myocardial	0.690569
Coronary	0.463368
Lipitor	0.438109
Aspirin	0.306466
Puffs	0.303476
Inferior	0.262048
MI	0.228691
CHF	0.210954
Catheterization	0.161087
...	...

Cuadro 5.1: *Los modelos de Regresión Logística ya entrenados se representan como una lista de características y sus coeficientes asociados. Esta tabla muestra las 13 características con mayores coeficientes para la categoría "Y" de la enfermedad Atherosclerotic CV disease (CAD). *INTERCEPT* representa la característica artificial incluida en el vector de características antes del entrenamiento para ajustar el valor de* β_0.

La tabla 5.2 muestra el acierto promedio (porcentaje de respuestas correctas) por enfermedad para la clasificación textual, utilizando validación cruzada de 4 subconjuntos sobre el conjunto de entrenamiento. Los buenos resultados obtenidos por el clasificador PMLC confirman que las clases están poco balanceadas, ya que a casi todos los historiales se les puede asignar la clase más probable con un alto grado de acierto. Esto es especialmente notorio para las enfermedades *Hypertriglyceridemia* y *Venous Insufficiency*, para las que el 98 % de los historiales fueron evaluados como de clase "U" (no se comenta en el historial nada acerca de la enfermedad). Sin embargo, el clasificador por regresión logística mejora los clasificadores Naïve Bayes y PMLC en el resto de enfermedades, siendo las diferencias en términos de acierto promedio de hasta 0,3 para clases mejor balanceadas como *CAD* o *Obesity* (datos en Tabla 5.2).

	PMLC	NB	LR
Obesity	0.6	0.7	0.9
Depression	0.86	0.86	0.89
Hypertriglyceridemia	0.98	0.98	0.97
Gallstones	0.85	0.85	0.89
OSA	0.86	0.86	0.92
Asthma	0.88	0.88	0.94
CAD	0.56	0.66	0.85
PVD	0.86	0.86	0.92
Gout	0.88	0.88	0.95
Diabetes	0.66	0.69	0.86
CHF	0.60	0.70	0.88
Venous Insufficiency	0.98	0.98	0.97
GERD	0.84	0.84	0.89
OA	0.85	0.85	0.89
Hypercholesterolemia	0.59	0.69	0.75
Hypertension	0.74	0.74	0.83

Cuadro 5.2: *Acierto promedio por enfermedad para la clasificación textual, utilizando validación cruzada 4-fold sobre el conjunto de entrenamiento. Se muestran los resultados obtenidos por los clasificadores PickMostLikelyCategory (PMLC), Naïve Bayes (NB) y Regresión Logística (LR).*

5.5.2 Evaluación comparativa de distintas representaciones de características.

Para evaluar la bondad de resultados obtenidos por el clasificador de regresión logística con cada una de las representaciones de características propuestas en la Sección 5.4.2, calculamos *precision*, *recall* y *f-measure* tal y como se definen en la sección 1.2.4.3. Dado que los ejemplos no están uniformemente distribuidos en las clases “Y”, “N”, “Q” y, en su caso, “U” para cada enfermedad, calculamos estas medidas promediadas por clase (*macro-averaged*) y por cada instancia del conjunto de test (*micro-averaged*).

La Tabla 5.3 muestra la *precision*, *recall* y *f-measure*, tanto macro-promediada como micro-promediada, para los tres clasificadores en el conjunto de test. Los resultados obtenidos por los distintos clasificadores confirman que las características adicionales que se introdujeron en la representación enriquecida (conjunto 2 en la tabla), son útiles para discernir el diagnóstico. Además, la utilización de un conjunto de características reducido con las características más informativas (conjunto 3 en la tabla), mejora la bondad de resultados para la evaluación intuitiva.

Type	Feature Set	Micro P	Macro P	Micro R	Macro R	Micro F	Macro F
Textual	1	0.9184	0.956	0.9184	0.4463	0.9184	0.4505
Textual	2	0.9233	0.5667	0.9233	0.4839	0.9233	0.5061
Textual	3	0.9288	0.7625	0.9288	0.4569	0.9288	0.4643
Intuitive	1	0.9019	0.9279	0.9019	0.5849	0.9019	0.5893
Intuitive	2	0.9061	0.9348	0.9061	0.5844	0.9061	0.5918
Intuitive	3	0.9244	0.9491	0.9244	0.5998	0.9244	0.6068

Cuadro 5.3: *Micro y Macro Precision (P), Recall (R) y F-measure (F) del clasificador de regresión logística en los datos de test para la clasificación textual e intuitiva utilizando las distintas representaciones de características propuestas en 5.4.2 (1: 5-gramas de caracteres; 2: léxico enriquecido con características adicionales; 3: Léxico reducido a los tokens más informativos).*

Tanto para la clasificación textual como intuitiva de historiales, los conjuntos 2 y 3 mejoran los resultados obtenidos por el conjunto 1, en términos de macro y micro *f-measure*. La mala clasificación de las instancias de las clases minoritarias ("Q" y "N" en la clasificación textual y "Q" en la clasificación intuitiva), explica la reducción dramática de la medida macro *f-measure* para estos clasificadores. Se necesitarían más instancias de estas clases para mejorar la bondad de clasificación para las mismas.

En la siguiente sección, analizamos en detalle la contribución que las distintas estrategias de formación de las características aportan a la bondad final del clasificador.

5.5.3 Análisis detallado de las distintas estrategias

En esta sección se discute con más detalle la efectividad de las estrategias adoptadas para la creación de las distintas representaciones para los historiales. También se justifican algunas de las decisiones tomadas durante el proceso de diseño de estas representaciones. Los resultados proporcionados en esta sección hacen referencia únicamente al conjunto de entrenamiento, puesto que las decisiones de diseño fueron adoptadas teniendo en consieración sólo este conjunto de instancias.

Efecto de la normalización de nombres de medicamentos

La Tabla 5.4 muestra la media y desviación estándar de la tasa de aciertos obtenida mediante validación cruzada en 4 subconjuntos considerando las siguientes situaciones respecto a la normalización de nombres de medicamentos:

A. Sin utilizar información adicional sobre los medicamentos.

B. Añadiendo una característica DISEASE_DRUG para cada medicamento (por ejemplo, añadiendo la característica DIABETES_DRUG con cada ocurrencia del token "lantus"), conforme lo explicado en la Sección 5.4.2.

C. Reemplazando la ocurrencia del medicamento por su correspondiente token DISEASE_DRUG (por ejemplo, reemplazando en el texto cada ocurrencia de "lantus" con DIABETES_DRUG).

Los resultados muestran que la adición de categorías de medicamentos (B) mejora el diagnóstico automático para las enfermedades Diabetes, Depresión e Hipertensión. Para el resto de enfermedades, los mejores resultados son aquellos obtenidos sin utilizar categorías de medicamentos (A). Por tanto, podemos concluir que estas tres enfermedades tienen asociado un conjunto de medicamentos mejor definido que los otros trastornos, que comparten un mayor número de medicamentos relacionados. A la luz de estos datos, el módulo de "Preprocesamiento" del sistema de clasificación representado en la Figura 5.1 sólo añade las categorías de medicamentos a aquellos fármacos relacionados con la diabetes, la depresión y la hipertensión.

Efecto del procesamiento de negaciones

La tabla 5.5 muestra un resumen de la tasa de acierto promedio antes y después del tratamiento de las negaciones, conforme lo explicado en la Sección 5.4.2. La tabla muestra sólo los resultados para diabetes, CAD, gota y depresión. Diabetes y CAD fueron seleccionadas por su alto número de historiales clasificados como "N" (un 2 % y un 2.6 %, respectivamente, del total del conjunto de entrenamiento). Gota y depresión fueron seleccionadas por su bajo número de historiales clasificados como "N" (inferior al 0.4 % en ambos casos).

Los resultados muestran que la adición de tokens negados produce un incremento en la tasa de acierto promedio para las enfermedades con mayor presencia de historiales clasificados como "N" y que la adición de estas características especiales no afecta negativamente en las enfermedades en las que apenas existe presencia de historiales clasificados como "N". Estos datos justifican que el procesamiento de negaciones haya sido implementado como parte del módulo "Preprocesamiento" del sistema de clasificación representado en la Figura 5.1.

Efecto de la redución del número de características.

La Figura 5.2 muestra la evolución de la tasa de acierto promedio para la tarea de clasificación intuitiva utilizando léxicos de distinto tamaño. La Figura 5.3 muestra la

Enfermedad	A	B	C
Diabetes	0.843 (0.031)	0.876 (0.017)	0.877 (0.018)
CHF	0.849 (0.017)	0.847 (0.021)	0.853 (0.018)
Asthma	0.931 (0.016)	0.929 (0.013)	0.930 (0.016)
CAD	0.821 (0.005)	0.807 (0.019)	0.804 (0.012)
Depression	0.864 (0.033)	0.878 (0.035)	0.876 (0.035)
Gallstones	0.864 (0.017)	0.867 (0.018)	0.845 (0.020)
GERD	0.895 (0.014)	0.888 (0.017)	0.893 (0.014)
Gout	0.952 (0.005)	0.946 (0.014)	0.945 (0.007)
Hypercholesterolemia	0.719 (0.042)	0.722 (0.038)	0.720 (0.046)
Hypertension	0.807 (0.014)	0.820 (0.022)	0.809 (0.011)
Hypertriglyceridemia	0.973 (0.015)	0.971 (0.019)	0.971 (0.019)
OA	0.872 (0.021)	0.874 (0.023)	0.875 (0.024)
Obesity	0.848 (0.018)	0.847 (0.031)	0.848 (0.023)
OSA	0.926 (0.011)	0.926 (0.008)	0.924 (0.011)
PVD	0.909 (0.014)	0.912 (0.013)	0.915 (0.012)
Venous Insufficiency	0.977 (0.008)	0.975 (0.006)	0.975 (0.006)
Promedio	0.8715	0.874	0.8723

Cuadro 5.4: *Efecto de la adición de categorías de fármacos en la tasa de acierto promedio por enfermedad. La tabla muestra la tasa de acierto promedio (con desviación estándar entre paréntesis) obtenida mediante validación cruzada 4-fold sin añadir categorías de fármacos (A), añadiendo categorías de fármacos (B) y reemplazando la ocurrencia del fármaco por su categoría asociada (C).*

Enfermedad	Sin añadir tokens negados	Añadiendo tokens negados
Diabetes	0.877 (0.018)	0.887 (0.021)
CAD	0.804 (0.012)	0.811 (0.024)
Gout	0.945 (0.007)	0.946 (0.010)
Depression	0.876 (0.035)	0.874 (0.048)

Cuadro 5.5: *Efecto del preprocesamiento de frases con negación en la tasa de acierto por enfermedad. La tabla muestra un resumen de la tasa de acierto promedio (con desv. estándar entre paréntesis) obtenida mediante validación cruzada 4-fold sobre los historiales sin añadir tokens negados y añadiendo tokens negados conforme a lo descrito en 5.4.2.*

evolución para la clasificación textual. En ambos casos el léxico reducido fue obtenido seleccionando las palabras de mayor *Information Gain*, conforme a lo descrito en la Sección 5.4.2. A la vista de los resultados se observa que la mejor tasa de acierto se obtiene utilizando léxicos muy pequeños e informativos.

El estudio detallado de los léxicos reducidos nos permite validar estrategias anteriores para enriquecer el léxico. Por ejemplo, el léxico reducido que contiene los $N = 5$ tokens más informativos para la clasificación textual contiene categorías de medicamentos (`diabetes_drug`, `depression_drug` and `hypertension_drug`) y algunos tokens negados (como, por ejemplo, `no_cri` para *hypertriglyceridemia*). La presencia de tokens negados aumenta significativamente conforme aumentamos el tamaño del léxico.

5.5.4 Intentos fallidos para mejorar la bondad de resultados.

En esta sección presentamos y discutimos algunos otros intentos para mejorar los resultados del clasificador que no fueron incluidos en el sistema final por no aportar mejora respecto a los resultados presentados anteriormente.

Incorporar la correlación entre enfermedades en el modelo.

Una de las hipótesis que surgió durante el diseño del sistema es que el emparejamiento de clasificadores para una pareja de enfermedades muy altamente correladas ayuda a mejorar los resultados respecto a la salida proporcionada por los dos clasificadores independientes. Para validar o refutar esta hipótesis, nos centramos en la clasificación textual y consideramos únicamente las dos clases mayoritarias “Y” y “U” (las evaluaciones “Q” y “N” se converten en “U”). Codificando “Y” como 1 y “U” como 0, calculamos la correlación entre cada pareja de enfermedades. La figura 5.4 muestra la distancia entre cada pareja de enfermedades utilizando distintas métricas.

Con el mismo objetivo, calculamos la matriz de probabilidades condicionadas para la clase “Y” de cada pareja de enfermedades (datos no mostrados). Un análisis detallado de esta matriz nos permitió observar casos interesantes, como que el padecer obesidad es muy probable si el paciente tiene OSA ($P(Obesity = Y|OSA = Y) = 0{,}86$), y muy poco probable cuando el paciente tiene PVD ($P(Obesity = Y|PVD = Y) = 0{,}21$).

Con este ejemplo en mente, pensamos que una buena validación para probar esta hipótesis sería añadir una nueva característica HAS_DISEASE_X para cada DISEASE_X que presenta el paciente, excepto para una enfermedad concreta, por ejemplo obesidad, y

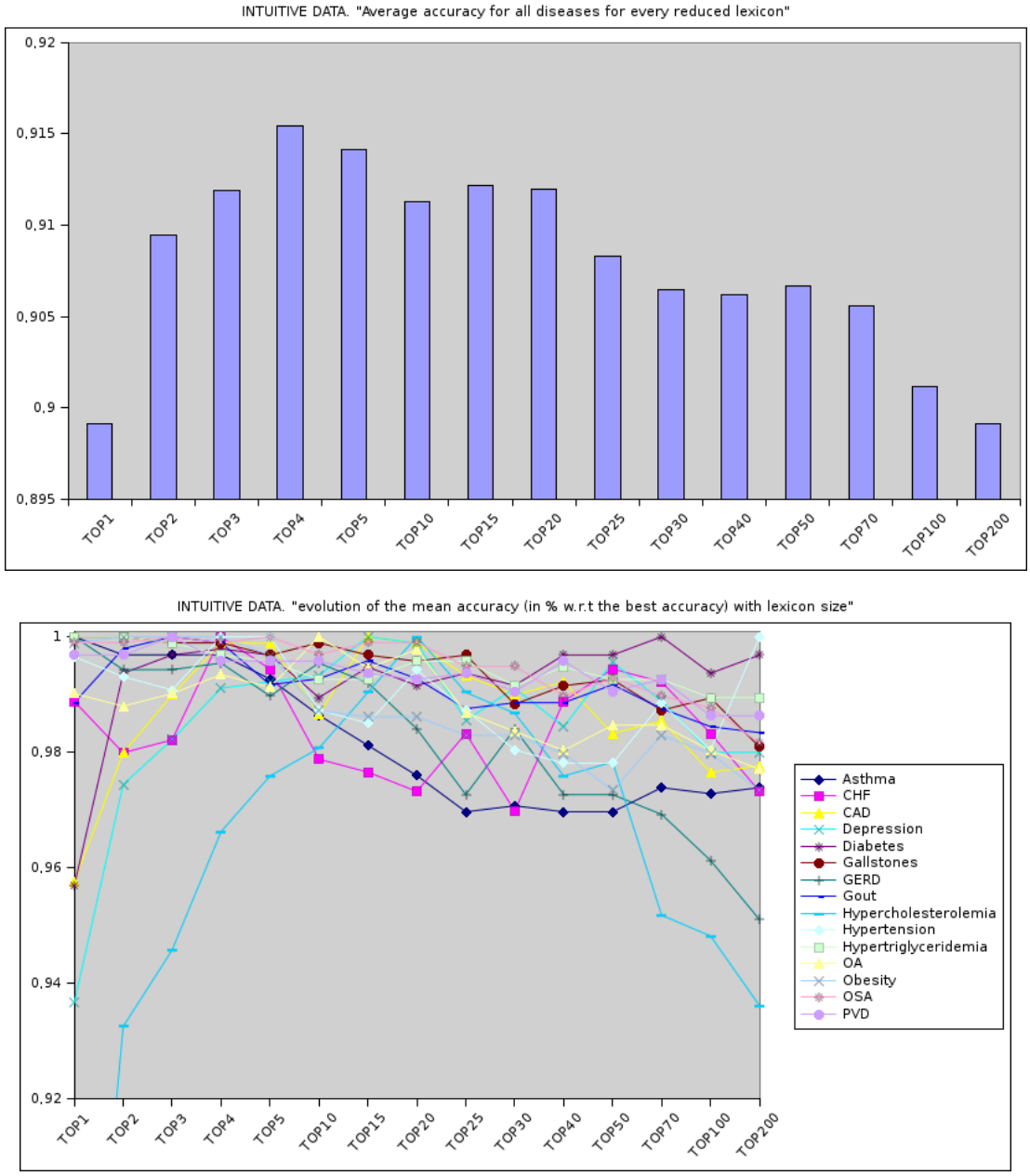


Figura 5.2: *Evolución de la tasa de acierto promedio para la clasificación intuitiva con léxicos reducidos.*

comprobar los resultados del clasificador para la evaluación de obesidad. Por ejemplo, si el paciente presenta OSA, CHF y Obesidad, a las características HAS_OSA y HAS_CHF se les asigna el valor 1.

Sin embargo, la tasa de acierto del clasificador de obesidad no presentó ninguna mejora utilizando estas características añadidas. Características como HAS_OSA, que consideramos que serían importantes para clasificar la categoría "Y" de obesidad, no

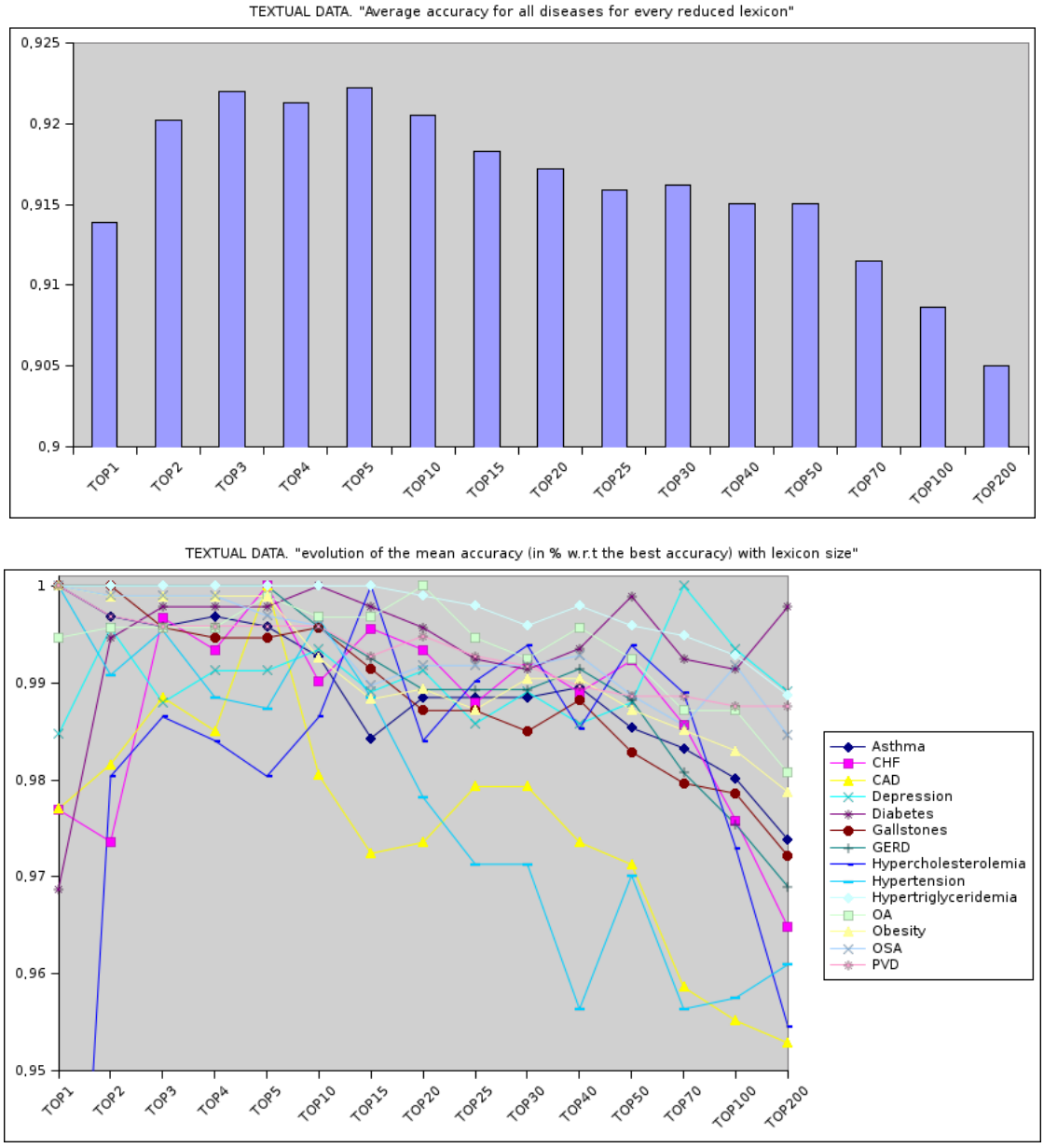
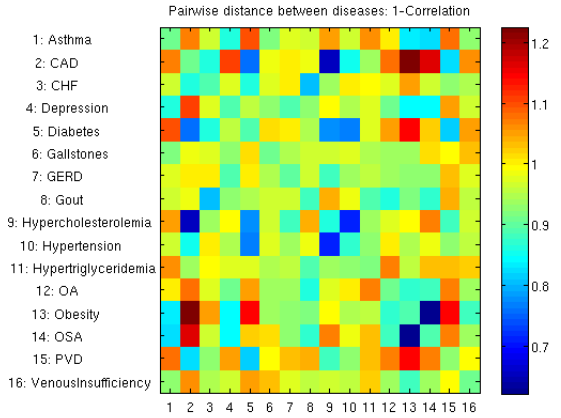


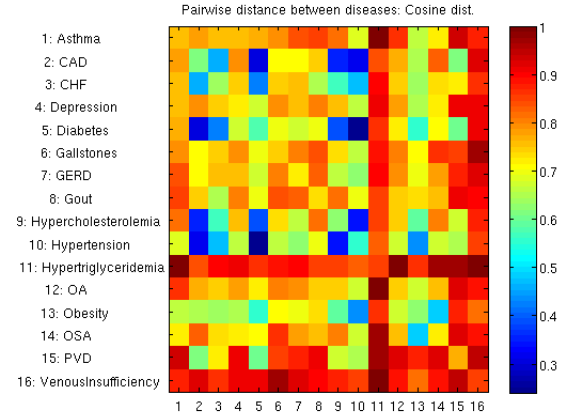
Figura 5.3: *Evolución de la tasa de acierto promedio para la clasificación textual con léxicos reducidos.*

son apenas tenidas en cuenta por el clasificador, a tenor del coeficiente β asociado a la misma.

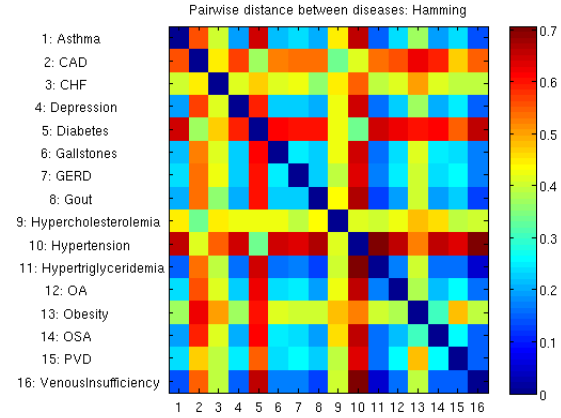
Como última prueba, decidimos crear una característica para cada tipo de clase y enfermedad: Y_DISEASE_X, U_DISEASE_X, Q_DISEASE_X, N_DISEASE_X, incluyendo estas características en la representación de los historiales para todas las enfermedades excepto para la que está siendo evaluada en un momento dado. Los resultados de las



(a) $1 - correlation$.



(b) *Cosine distance.*



(c) *Hamming distance.*

Figura 5.4: *Heatmaps mostrando distintas medidas de distancia entre cada pareja de enfermedades. Los valores de la diagonal, originalmente iguales a 0, se reemplazan en 5.4a y 5.4b por el valor promedio de las filas para permitir una mejor visualización de las matrices.*

Disease	Accuracy answers added	Accuracy no answers added
Asthma	0.929 (0.023)	0.929 (0.023)
CAD	0.774 (0.048)	0.763 (0.048)
CHF	0.854 (0.028)	0.856 (0.030)
Depression	0.866 (0.024)	0.866 (0.024)
Diabetes	0.805 (0.031)	0.805 (0.032)
Gallstones	0.847 (0.036)	0.847 (0.036)
GERD	0.843 (0.078)	0.843 (0.078)
Gout	0.921 (0.024)	0.921 (0.024)
Hypercholesterolemia	0.721 (0.053)	0.717 (0.053)
Hypertension	0.809 (0.053)	0.809 (0.053)
Hypertriglyceridemia	0.956 (0.016)	0.956 (0.016)
OA	0.859 (0.041)	0.859 (0.041)
Obesity	0.848 (0.044)	0.848 (0.044)
OSA	0.928 (0.017)	0.928 (0.017)
PVD	0.920 (0.012)	0.920 (0.012)
Venous Insufficiency	-	0.972 (0.014)

Cuadro 5.6: *Acierto promedio (con desv. típica entre paréntesis) de validación cruzada 4-fold para clasificación textual con las respuestas añadidas para las demás enfermedades (answers added) y sin ellas (no answers added). Las respuestas para otras enfermedades se añadieron como característica con la forma* Y_DISEASEX, U_DISEASEX, Q_DISEASEX, N_DISEASEX*, según corresponda.*

tabla 5.6 muestran que la mejora promedio es muy pequeña, en torno a 0,02 puntos en la tasa de acierto. Resultados similares se obtuvieron para la clasificación intuitiva 5.7.

5.6 Conclusiones.

En este capítulo hemos abordado la identificación automática de diagnósticos en textos clínicos utilizando técnicas de clasificación basadas en regresión logística y recursos terminológicos existentes. Los ensayos realizados sobre un corpus de historiales clínicos reales para pacientes tratados por obesidad y otras patologías relacionadas, muestran que la aproximación presentada ofrece buenos resultados en este ámbito. Además, dado que el sistema utiliza recursos terminológicos de ámbito general, éste puede ser fácilmente trasladado a cualquier otro dominio sin efectuar apenas ajustes, lo que supone una importante ventaja sobre otros sistemas existentes, diseñados *ad-hoc* para una patología concreta y que, por tanto, no tienen capacidad de generalización a otras patologías.

Disease	Accuracy answers added	Accuracy no answers added
Asthma	0.928 (0.016)	0.928 (0.016)
CAD	0.859 (0.036)	0.859 (0.036)
Depression	0.727 (0.048)	0.727 (0.048)
Diabetes	0.864 (0.019)	0.865 (0.018)
Gallstones	0.872 (0.040)	0.872 (0.040)
GERD	0.828 (0.030)	0.828 (0.030)
Gout	0.923 (0.024)	0.923 (0.024)
Hypercholesterolemia	0.738 (0.019)	0.745 (0.021)
Hypertension	0.836 (0.014)	0.836 (0.014)
Hypertriglyceridemia	0.928 (0.021)	0.928 (0.021)
OA	0.829 (0.023)	0.829 (0.023)
Obesity	0.847 (0.009)	0.848 (0.008)
OSA	0.933 (0.023)	0.933 (0.023)
PVD	0.907 (0.017)	0.907 (0.017)
Venous Insufficiency	-	0.894 (0.021)

Cuadro 5.7: *Acierto promedio (con desv. típica entre paréntesis) de validación cruzada 4-fold para clasificación intuitiva con las respuestas añadidas para las demás enfermedades (answers added) y sin ellas (no answers added).*

La extensa experimentación realizada permite mostrar la bondad del clasificador basado en regresion logística frente a clasificadores Naive Bayes y clasificadores que seleccionan la clase más probable. Estos ensayos muestran que a pesar del mal balanceo de las clases, el clasificador por regresión logística mejora los resultados de estos clasificadores.

De los tres conjuntos de características propuestos: 5-gramas, tokens enriquecidos con relaciones fármaco-enfermedad y negaciones, y léxico reducido con los tokens más informativos según el *information gain*; la utilización de un léxico reducido (de entre 4 y 10 tokens), obtenido seleccionando los tokens de mayor *information gain*, es el método más efectivo a la luz de los resultados mostrados para la experimentación efectuada.

Los valores macro y micro promediados de *precision*, *recall* y *f-measure* son obtenidos para cada conjunto de características, obteniéndose buenos valores para las medidas micro-promediadas (promediadas por instancia), pero mostrándose un dramático descenso en la f-measure macro-promediada (promediada por clase) para todos los clasificadores debido a la mala clasificación de instancias pertenecientes a las clases minoritarias ("Q" y "N" en la clasificación textual y "Q" en la clasificación intuitiva), especialmente en enfermedades con clases muy poco balanceadas. Para mejorar la

bondad de los resultados macro-promediados sería necesario contar con más instancias de estas clases.

Estudios detallados sobre la influencia de las distintas características en los resultados del clasificador final propuesto muestran que la adición de relaciones fármaco-enfermedad resulta efectiva para las enfermedades Diabetes, Depresión e Hipertensión, teniendo influencia negativa para el resto de enfermedades. Esto nos permite aventurar que estas tres enfermedades tienen un prototipo mejor definido de fármacos asociados, mientras que el resto de enfermades comparten un mayor número de fármacos relacionados. Además, los estudios sobre el efecto de la adición de tokens negados muestran que estas características son positivas para la clasificación de enfermedades con cierta presencia de instancias clasificadas como "N", sin perjudicar la bondad de clasificación de instancias de otras clases o enfermedades.

Además, se ha comprobado que a pesar de que ciertas categorias y enfermedades presentan dependencias entre sí (por ejemplo, padecer de obesidad parece probable si el paciente tiene OSA, $P(Obesity = Y|OSA = Y) = 0{,}86$, y poco probable cuando el paciente tiene PVD, $P(Obesity = Y|PVD = Y) = 0{,}21$); el encadenamiento de clasificadores para incorporar las predicciones para ciertas enfermedades en el cómputo de la clase para otras enfermedades, apenas ofrece mejora.

Parte IV

Conclusiones

Capítulo

6

Conclusiones y trabajo futuro.

6.1 Conclusiones

Esta memoria presenta varias aportaciones al estudio de los mecanismos de regulación celulares y la base genética de las enfermedades, mediante el análisis de datos producidos por tecnologías de microarrays de expresión y la información contenida en textos biomédicos y clínicos.

Clustering y biclustering en matrices de expresión genética.

La primera de las líneas de investigación presentadas propone distintas técnicas no supervisadas de clustering y biclustering para el análisis y la extracción de conocimiento de matrices de expresión genética obtenidas de microarrays. Las técnicas propuestas permiten solapamiento entre los grupos obtenidos, lo que las hace apropiadas para modelar la realidad biológica de que un mismo gen puede desempeñar distintas funciones y participar en diversos procesos biológicos conjuntamente con distintos grupos de genes.

Los métodos propuestos utilizan distintos criterios de optimización e implementan diversos paradigmas computacionales, desde el análisis de componentes principales, hasta algoritmos evolutivos o la lógica difusa.

En primer lugar, se han propuesto distintos algoritmos de clustering y biclustering basados en la función de optimización propuesta por Eisen *et al.* en el algoritmo Gene Shaving [135]. Este criterio busca maximizar tanto la coherencia del cluster como la varianza entre muestras. De este modo, los grupos de genes resultantes exhiben un comportamiento muy diferente entre muestras, lo que abre una vía para la caracterización de estos tipos de muestras y para el análisis de los procesos biológicos

y mecanismos regulatorios subyacentes que producen estas diferencias de comportamiento.

Los algoritmos propuestos han sido los siguientes:

- GA-Clustering: clustering utilizando algoritmos genéticos.
- EDA-Clustering: clustering utilizando algoritmos de estimación de distribuciones de probabilidad (EDA).
- Gene&Sample Shaving: biclustering basado en el análisis de componentes principales.
- EDA-Biclustering: biclustering utilizando algoritmos EDA.

Los resultados experimentales sobre dos *datasets* reales, uno que representa el ciclo celular de la levadura y otro que recoge muestras de distintos tipos de linfomas humanos, demuestran que los algoritmos de clustering y biclustering propuestos mejoran los resultados de Gene-Shaving en términos de calidad (GAP) y tamaño de los clusters/biclusters obtenidos. Para destacar la potencialidad del biclustering, la experimentación a este respecto se centró en el *dataset* de linfoma humano, que recoge un mayor número de condiciones y de alta heterogeneidad. Los biclusters obtenidos en este *dataset* mejoran ampliamente los resultados de Gene Shaving, encontrando patrones más refinados que incluso permiten discriminar ciertos tipos de cáncer del resto de muestras. El estudio detallado de los genes de estos grupos abre una puerta para la identificación de grupos funcionales y mecanismos regulatorios en estos organismos.

La segunda línea de investigación dentro de los métodos no supervisados para el análisis de matrices de expresión genética, propone un nuevo algoritmo denominado PSB (Possibilistic Spectral Biclustering) que identifica patrones potencialmente solapados con mínimo MSR y tamaño máximo. Este algoritmo utiliza técnicas espectrales de análisis de matrices y métodos posibilísticos de clustering. Los resultados experimentales obtenidos en *datasets* sintéticos y los *datasets* de la levadura y de linfomas humanos muestran que el PSB mejora los resultados de otros algoritmos de biclustering de similares características, como los algoritmos de Cheng & Church, FLOC y PLAID, en términos de calidad de los patrones obtenidos. Además, los resultados obtenidos por PSB presentan una mayor significación biológica que los del resto de algoritmos, ya que agrupan conjuntamente genes más relacionados respecto a su función biológica, a tenor de la información proporcionada por Gene Ontology. Del mismo modo, los resultados de PSB se ajustan mejor a la clasificación de muestras

proporcionada por Alizadeh *et al*. [25] para el *dataset de linfomas*, validando en cierto modo los resultados obtenidos y promoviendo que expertos en la materia efectúen un análisis más detallado de los patrones identificados.

Extracción de conocimiento de textos biomédicos.

La segunda línea de investigación desarrollada estudia la extracción de conocimiento de textos biomédicos, tanto de la literatura biomédica como de historiales clínicos, para identificar entidades de interés en estos textos (genes, proteínas, enfermedades, fármacos, etc.) y relaciones entre los mismos.

En esta línea, se ha propuesto BioNotate, una herramienta web de código libre para la anotación colaborativa de textos biomédicos. Esta herramienta viene a cubrir la necesidad creada por la ausencia de recursos que permitan sacar partido del extraordinario potencial de la comunidad científica y llevar a cabo esfuerzos distribuidos de anotación de textos. BioNotate es una herramienta flexible que permite implementar distintos esquemas de anotación, por lo que puede ser adaptada para satisfacer las demandas de distintos colectivos y grupos de investigación particulares. Por ejemplo, actualmente BioNotate está siendo empleada en la validación de interacciones entre genes y proteínas relacionados con el proyecto Autworks [2], y ha sido integrado en un entorno, denominado AutismNotate, para la anotación de genes de interés relacionados con el autismo (`http://bionotate.hms.harvard.edu/autism/`).

De este modo, BioNotate proporciona los recursos necesarios para crear un corpus de tamaño sustancial que permita el desarrollo y mejora de métodos de text-mining para la detección de entidades biomédicas y las relaciones entre ellas en textos biomédicos, a la vez que propone un marco que permite utilizar las anotaciones automáticas efectuadas por estos métodos para facilitar la anotación manual de los textos. Asimismo, en esta memoria se describe una extensión de BioNotate que incorpora representaciones y protocolos propios de la Web Semántica, para facilitar la difusión de las anotaciones en la comunidad científica y promover la integración del conocimiento generado con el disponible en otros recursos *online*.

En esta línea de investigación también se ha abordado la identificación automática de diagnósticos en textos clínicos utilizando técnicas de clasificación basadas en regresión logística y recursos terminológicos existentes. Los ensayos realizados sobre un corpus de historiales clínicos reales para pacientes tratados por obesidad y otras 15 patologías relacionadas, muestran que la aproximación presentada ofrece buenos resultados en este ámbito. Además, dado que el sistema utiliza recursos terminológicos

de ámbito general, éste puede ser fácilmente trasladado a cualquier otro dominio sin efectuar apenas ajustes, lo que supone una importante ventaja sobre otros sistemas existentes, diseñados *ad-hoc* para una patología concreta y que, por tanto, no tienen capacidad de generalización a otras patologías.

La extensa experimentación realizada permite mostrar la bondad del clasificador basado en regresion logística frente a clasificadores Naive Bayes y clasificadores que seleccionan la clase más probable. Estos ensayos muestran que a pesar del mal balanceo de las clases en el corpus disponible, el clasificador por regresión logística mejora los resultados de estos clasificadores.

6.2 Trabajo Futuro

A partir de las dos líneas de investigación desarrolladas, surgen numerosas posibilidades de investigación que nos proponemos acometer en un futuro próximo. A continuación describimos las de mayor interés.

Clustering y biclustering en matrices de expresión genética.

Respecto al análisis de la expresión genética con técnicas de Clustering y Biclustering, proponemos las siguientes líneas de actuación:

- Incorporar información sobre los tipos de muestras al proceso de clustering y biclustering sobre las matrices de expresión. Una línea de investigación interesante que se abre en este sentido sería la utilización de **Componentes Principales Supervisadas** [35] integradas en el algoritmo propuesto Gene-&-Sample Shaving, para maximizar la varianza entre las muestras de distintos tipos, de forma que el comportamiento de los genes de los agrupamientos obtenidos permita discriminar entre clases.
- Utilización de EDAs más avanzados, que tengan en consideración relaciones entre variables, para los algoritmos EDA-Clustering y EDA-Biclustering. Algunos de estos algoritmos son MIMIC [81] o COMIT [38], que consideran únicamente dependencias bi-variables; EMNA [192] es una aproximación basada en la estimación de una función de densidad multivariante normal en cada generación; BOA [246] y EBNA [193] incorporan técnicas para estimar una Red Bayesiana a partir de las soluciones prometedoras y utilizan esta red para generar nuevas soluciones.

- Validación de los clusters y biclusters de genes obtenidos utilizando metodologías más sofisticadas. Algunos estudios [213, 230] muestran que la definicion de clase funcional implementada por recursos como GO o KEGG, habitualmente empleados en la validación e interpretación biológica de los resultados del análisis de microarrays, no se corresponde exactamente con la co-expresión de los genes de la clase. En particular, [230] propose un escenario más realista en el que los módulos funcionales que no muestren co-expresión de los genes, sean excluidos del análisis estadístico para mejorar la potencia de predicción de cualquier test al realizar ajustes de múltiples hipótesis. Como trabajo futuro nos proponemos implementar estas metodologías de validación más sofisticadas para evaluar la significación biológica recogida por los métodos de clustering y biclustering propuestos.

Extracción de conocimiento de textos biomédicos.

Respecto a la anotación colaborativa de textos biomédicos utilizando BioNotate, existen distintas líneas de actuación que resultan prometedoras para ser implementadas en un futuro próximo:

- *Active Learning*. El *active learning* [279] es un campo del aprendizaje automático basado en la premisa de que un algoritmo de apendizaje supervisado puede obtener mejores resultados si es entrenado con un conjunto de instancias adecuado. Las técnicas de *active learning* permiten determinar qué objetos resultan más informativos o útiles para el entrenamiento del clasificador, minimizando, por tanto, el esfuerzo de anotación. BioNotate presenta un entorno idóneo para la aplicación de técnicas de *active learning* para la anotación de textos biomédicos (ver figura 6.1): mientras los anotadores humanos realizan los procesos de anotación de un conjunto de snippets, los sistemas automáticos de text-mining podrían ser ejecutados en el *back-end* de forma periódica para determinar, de forma transparente al anotador, qué snippets pendientes de anotación deben ser servidos a los anotadores para mejorar el rendimiento de los propios sistemas de text-mining.
- Adaptación de BioNotate a la plataforma Mechanical Turk de Amazon (AMT). Como ya se introdujo en la sección 1.4.1, la plataforma Mechanical Turk (AMT) de Amazon Web Services (AWS, `http://aws.amazon.com/`) es un servicio basado en esfuerzos distribuidos y comunitarios con una creciente aceptación en la comunidad científica. La utilización de AMT para la anotación colaborativa de

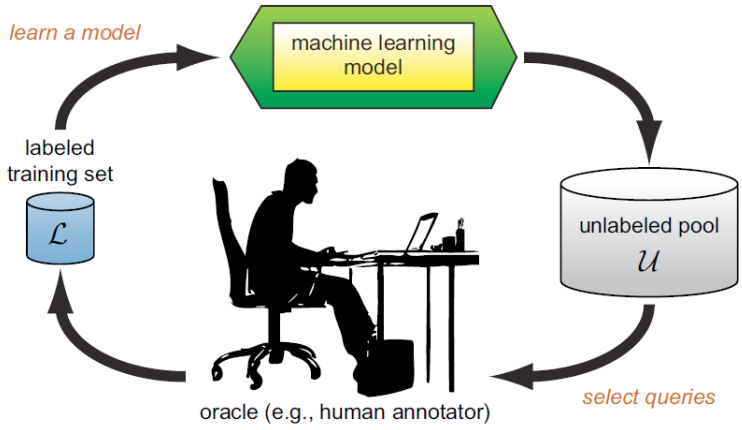


Figura 6.1: *Esquema del ciclo de aprendizaje con active learning [279].*

corpora para el entrenamiento de sistemas de ML es una opción que está siendo muy bien acogida por la comunidad, con numerosos artículos en esta línea publicados en el último año [259, 97, 60, 63]. La adaptación de BioNotate para que pueda ser utilizado como entorno de anotación en la plataforma AMT supone pues una línea de actuación muy interesante, debido a la todavía escasez de entornos de anotación potentes y adaptables a cualquier tipo de esquema de anotación para esfuerzos colaborativos en esta plataforma.

- Incorporación de herramientas más complejas de text-mining en BioNotate. Además de las herramientas para la identificación de entidades biomédicas (NER) incorporadas en BioNotate, como trabajo futuro nos proponemos incorporar herramientas más potentes de text-mining que permitan detectar relaciones entre las entidades en textos biomédicos. En esta línea también nos proponemos el desarrollo de nuestro propio método para identificar relaciones, basado en la definición de funciones kernels sobre los árboles de dependencias obtenidos por analizadores sintácticos para aprender a identificar el extracto de texto que soporta la existencia o no de una relación. La incorporación de ésta y otras herramientas de text-mining en BioNotate facilitarían la labor del anotador, proporcionándole candidatos identificados por estas herramientas para los hechos biológicos de interés en la tarea de anotación.

  En esta misma línea, nos planteamos adaptar BioNotate para hacer uso del meta-servidor de anotaciones automáticas derivado de las evaluaciones BioCreative [198], para obtener anotaciones sobre genes, especies, e interacciones proteína-proteína, consensuadas por distintos sistemas de text-mining, que

puedan ser manualmente corregidas por los anotadores.

- Adopción del protocolo DAS (*Distributed Annotation System*) [100] para distribuir las anotaciones a otras herramientas y servidores. Actualmente, existe un número creciente de servidores que implementan este protocolo para tomar datos de otras fuentes y distribuir la información que ellos mismos ofrecen. Por ejemplo, Ensembl y Uniprot distribuyen su información en formato DAS. Este protocolo permite obtener información de estos servidores de forma eficiente y a través de una API uniforme y estable.

Respecto a la extracción de conocimiento de historiales clínicos, nos planteamos la extensión de la metodología de clasificación basada en regresión logística a la identificación de otras entidades de interés como nombres de fármacos, dosis, frecuencia de administración de fármacos o síntomas mostrados por el paciente, entre otros.

Integración del conocimiento extraído de los microarrays con el de la literatura especializada

La integración de conocimiento de distintas fuentes de datos biológicas es utilizada en un creciente número de trabajos de investigación para ayudar a desvelar la base genética de las enfermedades [275].

Una de las posibilidades de integración que están siendo valoradas consistiría en construir, siguiendo los pasos del proyecto Autworks (4.5), redes genéticas con nodos representando genes que han sido asociados a una enfermedad determinada, y arcos entre nodos representando relaciones entre los genes, extraídas tanto de la literatura biomédica como del análisis de microarrays. De este modo, la evidencia de que existe interacción entre dos genes se refuerza si los dos genes son identificados como parte de un mismo módulo funcional (cluster o bicluster) y existen trabajos previos que han constatado tal relación. La ponderación de los arcos de la red se correspondería con la cantidad y fiabilidad de las evidencias detectadas. Estas redes pueden ser analizadas en detalle para intentar descubrir la base genética de enfermedades [166, 151, 223].

Chapter

7

Conclusions and future work

7.1 Conclusions

This dissertation provides several novel contributions to the decomposition and ultimate interpretation of the genetic basis of disease based on the analysis of high-throughput gene expression data and biomedical texts.

Clustering and biclustering gene expression matrices.

We accomplished the analysis of gene expression data by using several clustering and biclustering algorithms for the identification of potential functional modules. As a gene may play more than one biological role in conjunction with distinct groups of genes, the algorithms we proposed are non-exclusive (i.e. they allow potential overlapping among the obtained clusters).

The proposed methods explore different optimization criteria and make use of several computational paradigms, ranging from principal component analysis to evolutionary algorithms and fuzzy technology.

Firstly, we have proposed several clustering and biclustering methods based on the optimization criteria proposed by Hastie *et al.* in their Gene Shaving algorithm. This criterion maximizes both the cluster coherence and the between-sample variance. Resultant groups of genes exhibit very different behaviour across samples, leading to further research on the biological processes which may produce these differences. We have described the following algorithms:

- GA-Clustering: clustering based on genetic algorithms.
- EDA-Clustering: clustering based on estimation of distribution algorithms (EDAs).

- Gene&Sample Shaving: biclustering based on principal component analysis.
- EDA-Biclustering: biclustering based on EDAs.

Experimental results on *S. cerevisiae* cell cycle expression data and on a human lymphoma dataset show that the proposed methods outperform Gene Shaving in terms of quality (GAP) and size of revealed patterns. The human lymphoma dataset contains a greater number of heterogeneous conditions (9 different types of lymphoma and healthy tissues), becoming a good test-case for showing the potential of biclustering. The biclusters obtained for this dataset clearly improve the results of Gene Shaving, finding detailed patterns which allow to discriminate certain types of lymphoma from other samples. A detailed study of the genes of these groups is needed to shed light on the identification of functional groups and regulatory mechanisms in these organisms.

Secondly, we have proposed a new method called Possibilistic Spectral Biclustering algorithm (PSB), to identify potentially-overlapping patterns of minimum MSR and maximum size. This algorithm is based on Fuzzy Technology and Spectral Clustering. We tested this method on syntetic data and on real data (S. cerevisiae cell cycle expression data and the human lymphoma dataset), outperforming the results obtained by other biclustering algorithms, such as the one proposed by Cheng & Church, FLOC and PLAID. Moreover, the results obtained by PSB show greater biological significance than the rest of the algorithms, as more closely related genes are clustered together based on their biological function, according to the information provided by the Gene Ontology. Similarly, PSB results fit better the classification of samples provided by Alizadeh *et al.* [25] for the lymphoma dataset.

Knowledge extraction from biomedical texts.

Our second line of research describes our efforts towards the extraction of knowledge from biomedical texts, namely the biomedical literature and clinical discharge summaries. Particularly, we focused on the identification of entities of interest in these texts (genes, proteins, diseases, etc.) and relationships among them.

Along this line of research, we propose BioNotate, a web-based open-source collaborative annotation tool for biomedical text. The creation of this tool is hampered by the absence of a resource for launching a distributed annotation effort which can be suited to different needs, i.e. adapted to different annotation schemas. For example, BioNotate has been used for validating gene-gene interactions in an autism gene network from the Autworks project [2], and it has also been used as annotation

platform for annotating genes in an Autism Consortium-related annotation effort (the BioNotate-Autism initiative, `http://bionotate.hms.harvard.edu/autism/`).

BioNotate provides the community with an annotation tool to harness the great collaborative power of biomedical community over the internet to create substantially sized corpora as a baseline for research on biomedical text-mining, while simultaneously using automated tools to facilitate the manual curation of text. Furthermore, we described an extension which allows curated facts to be published and shared with the community in RDF format, and conveniently accessed and browsed with the provided Linked Data front-end. This guarantees an easier integration of the annotated facts with other community resources.

We have also proposed a methodology for rapidly tailoring a logistic regression classifier to the automatic identification of diagnoses in clinical discharge summaries. We applied this methodology to a corpus of discharge summaries which correspond to patients who were evaluated for obesity and 15 of its best represented co-morbidities. Our results showed high performance (average F-micro 0.92) for the task of classifying each discharge summary with the correct diagnoses. Furthermore, since this system only uses general-purposes terminological resources, it could be easily adapted to any other domain. This constitutes an important advantage over other existing systems, which are strongly tailored to specific domains and ad-hoc patterns so they poorly generalize across diseases or diagnoses. A comparative evaluation of the logistic regression classifier together with a Naive Bayes classifier and a classifier which selects the most likely class was conducted, showing that despite the unbalanced constitution of the corpus, the logistic regression classifier outperforms the other classifiers.

7.2 Future work

The work and contributions proposed in this dissertation may give rise to several lines of future research. Here we briefly describe the most interesting ones.

Clustering and biclustering gene expression data.

Regarding the analysis of gene expression data by means of clustering and biclustering, we propose the following lines of research:

- Incorporate information about the samples to the clustering and biclustering processes. An interesting line of research in this direction is the use of **Supervised Principal Components** [35] embedded into the proposed Gene-&-Sample Shaving algorithm. This would allow to maximize the variance between

samples of different types, so that the behavior of the genes in the resultant clusters would discriminate between the classes of samples.

- Using advanced EDAs that take into account relationships between variables, for the algorithms EDA-Clustering and EDA-Biclustering. Some of these EDAs are MIMIC [81] or COMIT [38], which only consider bivariate dependencies; Emna [192] is an approximation based on the estimation of a multivariate normal density function in each generation or BOA [246] and EBNA [193] which incorporate techniques for inferring a Bayesian network from the most promising solutions and use the network to generate new solutions.
- Validation of the obtained gene clusters and biclusters using more sophisticated methodologies. Some studies [213, 230] show that the definition of functional classes in resources such as GO or KEGG (which are frequently used for the validation and interpretation of microarray analysis) does not always correspond to co-expressed sets of genes. Particularly, [230] propose a more realistic scenario in which the functional modules that do not show co-expression of the genes, are excluded from the functional analysis to increase the power of any test in the adjustment for multiple testing. Future work includes the implementation of these more sophisticated validation methodologies to assess the biological significance of the obtained results.

Extracting knowledge from biomedical texts.

Regarding the collaborative annotation of biomedical texts using BioNotate, there are different lines of research which can be further studied in the near future:

- *Active Learning*. The *active learning* [279] is a field of machine learning based on the assumption that a supervised learning algorithm can get better results if it is trained with a set of appropriate instances. Active learning techniques allow to determine the most useful or informative instances for training a classifier, thus minimizing the annotation effort. BioNotate provides a suitable environment for the application of active learning (see Figure 6.1): while the human annotators curate a set of snippets, text-mining tools can be run in the back-end to determine what snippets pending annotation must be served to the users next, in order to improve performance of the text-mining systems.
- Adaptation of BioNotate to the Amazon Mechanical Turk (AMT) platform. As introduced in section 1.4.1, the use of the AMT plaform from Amazon Web

Services (AWS `http://aws.amazon.com/`) to perform collaborative annotation efforts on textual corpora, is increasing its popularity amongst the community, with several publications addressing this issue in the last year [259, 97, 60, 63]. Adapting BioNotate to be used as annotation platform in AMT services is a very promising line of research due to the lack of flexible and powerful annotation environments in this platform.

- Addition of more complex text-mining tools to BioNotate. Apart from the NER tools included in BioNotate as described in section 4.6.3, we plan to add more powerful text-mining tools for detecting NE and relationships between them in biomedical texts. Of particular interest is the adaptation of BioNotate to make use of the meta-server derived from the BioCreative evaluations [198] to obtain consensus annotations on genes, species and protein-protein interactions made by different text-mining tools, which can be further corrected by human curators. We also plan to develop our own method to identify relationships between entities of interest, by making use of kernel functions defined on the syntactic structures obtained by full-parsers.
- Adoption of DAS (*Distributed Annotation System*) [100] to distribute the annotated facts to other tools and servers. Currenty, there is a growing number of servers that implement this protocol to import data from other sources and distribute the data they provide. For example, Ensembl and Uniprot distribute their information in DAS format. This protocol enhaces the inter-operability between different servers through a uniform and stable API.

Regarding the extraction of knowledge from medical records, we plan to extend the methodology for rapid prototyping of a classifier based on logistic regression to the identification of other entities of interest in discharge summaries such as drugs, dosages, frequency of drug administration or symptoms, among others.

Knowledge integration

The integration of knowledge from various biological data sources is used by an increasing number of works to help unveiling the genetic basis of disease [275].

One possibility for the integration of knowledge that we are evaluating is the creation of gene networks with nodes representing genes that are associated to a particular disease, and arcs between nodes representing interactions/associations drawn from both the biomedical literature and microarray data analysis. The evidence of interaction between two genes will be strengthened if they are identified as part of

the same functional module (cluster or bicluster) and there are previous works that report such a relationship. The amount and reliability of the evidences supporting an interaction would determine the weight of the corresponding arc. These networks can be further analyzed to uncover the etiology of diseases [166, 151, 223].

Parte V

Publicaciones

Capítulo

8

Trabajos publicados

Publicaciones derivadas del trabajo expuesto en la memoria

Relacionados con el desarrollo de esta memoria, se han publicado los siguientes trabajos como autor principal:

Publicaciones en revistas

- **C. Cano**, L. Adarve, J. López, A. Blanco. (2007) *Possibilistic Approach for Biclustering Microarray Data*. Computers in Biology and Medicine (ISSN: 0010-4825). Elsevier. Vol 37/10 , pgs 1426-1436.
- **C. Cano**, F.J. Lopez, F. Garcia, A. Blanco. (2009) *Intelligent system for the analysis of microarray data using principal components and estimation of distribution algorithms*. Expert Systems With Applications, Vol. 36, pgs. 4654-4663.
- **C. Cano**, T. Monaghan, A. Blanco, D.P. Wall, L. Peshkin. (2009) *Collaborative text-annotation resource for disease-centered relation extraction from biomedical text*. Journal of Biomedical Informatics, Vol. 42, pgs. 967-977.
- **C. Cano**, A. Blanco, L. Peshkin. (2009) *Automated Identification of Diagnosis and Co-morbidity in Clinical Records*. Methods of Information in Medicine, Vol. 48, N. 6, pgs. 546-551.
- **C. Cano**, A. Blanco, F. García, F.J. López. (2006) *Evolutionary Algorithms for Finding Interpretable Patterns in Gene Expression Data*. International Journal of

Computer Science and Information Systems (ISSN 1646-3692), No. 2, pgs 88-99.

- F.J. Esteban, **C. Cano**, I. de la Haza, A. Cano-Ortiz, N. V. Mendizabal, J. Goñi, J.A. Horcajadas (2008) *Análisis bioinformático de datos: aplicación en microarrays*. Cuadernos de Medicina Reproductiva, Vol. 14 , N.1, pgs 87-96.

Publicaciones en actas de congresos

- **C. Cano**, A. Blanco, F. García, F.J. López. (2006) *Evolutionary Algorithms for Finding Interpretable Patterns in Gene Expression Data*. Proceedings of the IADIS International Conference. Applied Computing 2006. (ISSN: 972-8924-09-7). Número: 1. 25-28 Febrero de 2006. San Sebastián. España.
- **C.Cano**, F.L. Adarve, F. García, F.J. López, A. Blanco (2007) *Non-supervised identification of gene regulatory modules by possibilistic biclustering of microarray data* Proceedings of the 11th International conference on Cognitive and Neural Systems. 16-19 Mayo, Boston, MA, USA.
- **C. Cano**, S. Blanco, F. García, A. Blanco. (2008) *Max-variance Clustering and Biclustering of Microarray Data*. Proceedings of the 12th Int. Conf. on Information Processing and Management of Uncertainty in Knowledge-based systems (IPMU'08), ppgs. 690-697. Junio 22-27, 2008. Málaga, España.
- **C. Cano**, L. Peshkin, B. Carpenter, B. Baldwin. (2008). *Regularized Logistic Regression for Clinical Record Processing*. Proceedings of the 2nd i2b2 workshop on Challenges in Natural Language Processing for Clinical Data. 7-8 Noviembre 2008, Washington, D.C., USA.
- **C.Cano**, A. Blanco, D.P. Wall, L. Peshkin (2009). *BioNotate: una herramienta web para la anotación colaborativa de textos biomédicos.* Actas de las I Jornadas Andaluzas de Informática, pps. 103-108. Septiembre 2009. Málaga, España.

Otras publicaciones relacionadas con el trabajo expuesto en la memoria

Relacionados con el tema tratado en esta memoria, también se ha colaborado en las siguientes publicaciones:

Publicaciones en revistas

- F. J Lopez, A. Blanco, F. Garcia, **C. Cano**, A. Marin (2008) *Fuzzy Association Rules for Biological Data Analysis: a case study on yeast*. BMC Bioinformatics, 9:107.
- F. García, FJ. López, **C. Cano**, A. Blanco (2009) *FISim: a new similarity measure between transcription factor binding sites based on the fuzzy integral*. BMC Bioinformatics, 10:224.
- Capítulo de libro: FJ Lopez; A Blanco; F García; **C. Cano** (2009) *Extracting Biological Knowledge by Association Rule Mining*, publicado en *Data mining in biomedicine using ontologies* (ISBN-13: 978-1596933705), ppg. 133-161.

Publicaciones en Actas de Congresos

- J. Sainz, A. Barroso, A. Blanco, F.García, **C. Cano**, A. Concha. (2005) *Gene Expression Profiling in Mouse Embryonic Stem Cells*. Simposio Internacional sobre Nuevos Avances en Medicina Reproductiva. Valencia. España.
- J. Sainz, A. Barroso, A. Blanco, **C. Cano**, F.García, A. Concha. (2005) *Microarray Analysis of Mouse Embrionic Stem Cells*. 32 Symposium Internacional Fertilidad 2005. Barcelona. España.
- F. García, F.J. López, **C. Cano**, A. Blanco. (2006) *An Ontology-Driven Similarity Providing Reliable Protein Family Recognition* . Proceedings of the IADIS International Conference. Applied Computing 2006 (ISSN: 972-8924-09-7). Número: 1, pgs 649-654. 25-28 Febrero de 2006. San Sebastián. España.
- S.Blanco, **C. Cano**, F.J. López, A. Blanco. (2007) *SCEPG: Un método de Biclustering para Datos de Microarrays.*. Actas del V Congreso Español sobre Metaheurísticas, Algoritmos Evolutivos y Bioinspirados MAEB2007 (ISBN: 978-84-690-3470-5). pgs 427-434. 14-16 Febrero de 2007. Tenerife. España.

- S.Blanco, **C. Cano**, F.J. López, A. Blanco. (2007) *SCEPGG: A Method for biclustering Microarray Data.* Proceedings of the IADIS International Conference Applied Computing 2007 (ISBN: 978-972-8924-30-0). pgs 493-498. 18-20 Febrero de 2007. Salamanca. España.
- P. Bueno, C. Olmedo, A. Comino, L. Hassan, K. Muffak, **C. Cano**, M. Serradilla, A. García-Navarro, A. Mansilla, J. Villar, D. Garrote, A. Blanco, J.A. Ferrón (2007). *Microarray Study of Gene Expression Profile in Liver Transplant Recipients With a Diagnosis of Hepatitis C Virus.*. Proceedings of the 13rd Congress of the European Society of Organ Transplantation (ESOT 2007), publicado en Transplant International, Vol. 20. 29 Sep- 3 Oct, 2007. Praga, República Checa
- L. Hassan, P. Bueno, C. Olmedo, A. Comino, **C. Cano**, I. Ferrón-Celma, K. Muffak, M. Serradilla, A. García-Navarro, A. Mansilla, J. Villar, D. Garrote, A. Blanco, J.A. Ferrón. (2007) *Gene expression profiling in Living transplant recipients*. Proceedings of the 13th International Conference of the Liver Transplantation Society, publicado en Liver Transplantation, Vol. 13, No. 6. Junio 20-27, 2007. Rio de Janeiro, Brasil.
- A. Irigoyen, C. Olmedo, A. Comino, **C. Cano**, J. Valdivia, B. Jimenez-Rubiano, B. Gonzalez-Astorga, J.Delgado, A. Blanco, P. Bueno. (2008) *Microarray study of gene expression profile in peripheral blood samples from lung cancer patients*. Proceedings of the ASCO-NCI-EORTC Annual Meeting on Molecular Markers in Cancer. Oct. 30-Nov. 1 2008. Florida, EEUU.
- FJ López; A. Blanco; F. García; **C. Cano**; A. Marín. (2009). *A Fuzzy Approach For The Study Of Functional And Structural Features Of The Yeast Genome*. Proceedings of the 13th Evolutionary Biology Meeting. Marsella, Francia
- FJ. López; **C. Cano**; F. García; A. Blanco. (2009). *A Fuzzy Approach For Studying Combinatorial Regulatory Actions Of Transcription Factors In Yeast*. Proceedings of the 10th International Conference on Intelligent Data Engineering and Automated Learning (IDEAL'09). Lecture Notes in Computer Science. Springer.
- F. García, FJ López, **C.Cano**, A. Blanco (2009). *Study of Fuzzy Resemblance Measures for DNA Motifs*. Proceedings of the 2009 IEEE International Conference on Fuzzy System. Agosto 2009, Korea.
- A. Irigoyen, C. Olmedo, J. Valdivia, A. Comino, **C. Cano**, R. Luque, V. Conde, J. Delgado, A. Blanco, P. Bueno. (2009). *Microarray study of gene expression profile in peripheral blood samples from lung cancer patients before and after erlotinib*

treatment. Proceedings of the ASCO-NCI-EORTC Annual Meeting on Molecular Markers in Cancer. 15-17 Oct. 2009. Bruselas, Bélgica.

- P. Palma, C. Olmedo, **C.Cano**, R. Conde, A. Comino, M. Cuadros, I. Segura, E. Coll, E. González-Flores, R. del Moral, A. Blanco, P. Bueno, J.A. Ferrón. (2009). *Gene expression profile in rectal cancer: impact of 5-FU based neoadjuvant treatment.*. Proceedings of the Annual Meeting of the European Society of Coloproctology. Praga, República Checa. 2009.

Bibliografía

[1] AIMed Corpus. ftp://ftp.cs.utexas.edu/pub/mooney/bio-data/.

[2] Autworks. http://autworks.hms.harvard.edu/.

[3] BioCreAtIvE-I Task 1A Corpus Enriched with Annotations for Interactions Between Genes/proteins.
http://www2.informatik.hu-berlin.de/ hakenber/corpora/.

[4] DepGENIA Corpus. http://www.ifi.unizh.ch/cl/kalju/download/depgenia/.

[5] Fetch prot corpus.
http://www.sics.se/humle/projects/fetchprot/Corpus/Release20051011/.

[6] GeneCards. http://genecards.org/.

[7] Genic Interaction Extraction Challenge LLL Workshop 05.
http://genome.jouy.inra.fr/texte/LLLchallenge/.

[8] Genome sequencing projects at the wellcome trust sanger institute.

[9] Hiv-1 human protein interaction database.
http://www.ncbi.nlm.nih.gov/RefSeq/HIVInteractions/index.html.

[10] iAnnotate.
http://www.dbmi.columbia.edu/ cop7001/iAnnotateTab/iannotate.htm.

[11] Knowtator. http://knowtator.sourceforge.net.

[12] OMIM. http://www.ncbi.nlm.nih.gov/omim/.

[13] PennBioIE Corpus. http://bioie.ldc.upenn.edu/.

[14] WordFreak. http://wordfreak.sourceforge.net.

[15] Yapex Corpus. http://www.sics.se/humle/projects/prothalt/.

[16] The human genome (nature entire issue), 2001.

[17] Building on the dna revolution (science entire issue), 2003.

[18] Biocreative II.5, 2009. http://www.biocreative.org/events/biocreative-ii5.

[19] P. Agarwal and DB Searls. Literature mining in support of drug discovery. *Briefings in bioinformatics*, 9(6):479–492, 2008.

[20] J.S. Aguilar-Ruiz and F. Divina. Evolutionary biclustering of microarray data. *Applications of evolutionary computing, proceedings lecture notes in computer science*, 3449:1–10, 2005.

[21] S.T. Ahmed, D. Chidambaram, H. Davulcu, and C. Baral. Intex: A syntactic role driven protein-protein interaction extractor for bio-medical text. In *Proceedings of the ACL-ISMB Workshop on Linking Biological Literature, Ontologies and Databases: Mining Biological Semantics*, pages 54–61, 2005.

[22] A. Airola, S. Pyysalo, J. Bjorne, T. Pahikkala, F. Ginter, and T. Salakoski. All-paths graph kernel for protein-protein interaction extraction with evaluation of cross-corpus learning. *BMC bioinformatics*, 9(Suppl 11):S2, 2008.

[23] F. Al-Shahrour, R. Díaz-Uriarte, and J. Dopazo. FatiGO: a web tool for finding significant associations of Gene Ontology terms with groups of genes. *Bioinformatics*, 20(4):578, 2004.

[24] C. Alfarano, CE Andrade, K. Anthony, N. Bahroos, M. Bajec, K. Bantoft, D. Betel, B. Bobechko, K. Boutilier, E. Burgess, et al. The biomolecular interaction network database and related tools 2005 update. *Nucleic acids research*, 33(Database Issue):D418, 2005.

[25] A.A. Alizadeh, M.B. Eisen, R.E. Davis, C. Ma, I.S. Lossos, A. Rosenwald, J.C. Boldrick, H. Sabet, T. Tran, X. Yu, et al. Distinct types of diffuse large B-cell lymphoma identified by gene expression profiling. *Nature*, 403(6769):503–511, 2000.

[26] J. Allen. *Natural language understanding*. Benjamin/Cummings, 1995.

[27] N.S. Altman and J. Hua. Extending the loop design for two-channel microarray experiments. *Genetics Research*, 88(03):153–163, 2007.

[28] S. Ananiadou and J. McNaught. *Text mining for biology and biomedicine*. Artech House Publishers, 2006.

[29] R.K. Ando. BioCreative II gene mention tagging system at IBM Watson. In *Proceedings of the Second BioCreative Challenge Evaluation Workshop*, pages 101–103, 2007.

[30] A.V. Antonov, I.V. Tetko, V.V. Prokopenko, D. Kosykh, and H.W. Mewes. A web portal for classification of expression data using maximal margin linear programming. *Bioinformatics*, 20(17):3284, 2004.

[31] R. Armañanzas, I. Inza, R. Santana, Y. Saeys, J.L. Flores, J.A. Lozano, Y. Van de Peer, R. Blanco, V. Robles, C. Bielza, et al. A review of estimation of distribution algorithms in bioinformatics. *BioData mining*, 1:6, 2008.

[32] M. Ashburner, C.A. Ball, J.A. Blake, D. Botstein, H. Butler, J.M. Cherry, A.P. Davis, K. Dolinski, S.S. Dwight, J.T. Eppig, et al. Gene Ontology: tool for the unification of biology. *Nature genetics*, 25(1):25–29, 2000.

[33] R. Avogadri and G. Valentini. Fuzzy ensemble clustering based on random projections for DNA microarray data analysis. *Artificial Intelligence in Medicine*, 45(2-3):173–183, 2009.

[34] A.M. Bagirov and K. Mardaneh. Modified global k-means algorithm for clustering in gene expression data sets. In *Proceedings of the 2006 workshop on Intelligent systems for bioinformatics-Volume 73*, page 28. Australian Computer Society, Inc., 2006.

[35] E. Bair, T. Hastie, D. Paul, and R. Tibshirani. Prediction by supervised principal components. *Journal of the American Statistical Association*, 101(473):119–137, 2006.

[36] A. Bairoch, R. Apweiler, C.H. Wu, W.C. Barker, B. Boeckmann, S. Ferro, E. Gasteiger, H. Huang, R. Lopez, M. Magrane, et al. The universal protein resource (UniProt). *Nucleic Acids Research*, 33(Database Issue):D154, 2005.

[37] P. Baldi, S. Brunak, Y. Chauvin, C.A.F. Andersen, and H.Ñielsen. Assessing the accuracy of prediction algorithms for classification: an overview. *Bioinformatics*, 16(5):412, 2000.

[38] S. Baluja, S. Davies. Combining multiple optimization runs with optimal dependency trees. Technical report, Computer Science Department, Carnegie Mellon University, 1997.

[39] S. Bandyopadhyay, A. Mukhopadhyay, and U. Maulik. An improved algorithm for clustering gene expression data. *Bioinformatics*, 23(21):2859, 2007.

[40] Z. Bar-Joseph, E.D. Demaine, D.K. Gifford, N. Srebro, A.M. Hamel, and T.S. Jaakkola. K-ary clustering with optimal leaf ordering for gene expression data. *Bioinformatics*, 19(9):1070, 2003.

[41] C. Baral, G. Gonzalez, A. Gitter, C. Teegarden, and A. Zeigler. CBioC: beyond a prototype for collaborative annotation of molecular interactions from the literature. In *Computational Systems Bioinformatics: CSB2007 Conference Proceedings, Volume 6: University of California, San Diego, USA, 13-17 August 2007*, page 381. Imperial College Pr, 2007.

[42] S. Barkow, S. Bleuler, A. Prelic, P. Zimmermann, and E. Zitzler. BicAT: a biclustering analysis toolbox. *Bioinformatics*, 22(10):1282, 2006.

[43] W.A. Baumgartner Jr, K.B. Cohen, L.M. Fox, G. Acquaah-Mensah, and L. Hunter. Manual curation is not sufficient for annotation of genomic databases. *Bioinformatics*, 23(13):i41, 2007.

[44] N. Belacel, M. Cuperlovic-Culf, M. Laflamme, and R. Ouellette. Fuzzy J-Means and VNS methods for clustering genes from microarray data. *Bioinformatics*, 20(11):1690, 2004.

[45] F. Belleau, M.A. Nolin, N. Tourigny, P. Rigault, and J. Morissette. Bio2RDF: Towards a mashup to build bioinformatics knowledge systems. *Journal of biomedical informatics*, 41(5):706–716, 2008.

[46] L. Bernardi, E. Ratsch, R. Kania, J. Saric, I. Rojas, JH Park, BR Schatz, C. Blaschke, A. Valencia, and C.Ñedellec. Mining information for functional genomics. *IEEE Intelligent Systems*, 17(3):66–80, 2002.

[47] J.C. Bezdek. *Pattern recognition with fuzzy objective function algorithms*. Kluwer Academic Publishers Norwell, MA, USA, 1981.

[48] A. Bhattacharya and R.K. De. Divisive Correlation Clustering Algorithm (DCCA) for grouping of genes: detecting varying patterns in expression profiles. *Bioinformatics*, 24(11):1359, 2008.

[49] A. Bhattacharya and R.K. De. Bi-correlation clustering algorithm for determining a set of co-regulated genes. *Bioinformatics*, 25(21):2795, 2009.

[50] J. Bjorne, J. Heimonen, F. Ginter, A. Airola, T. Pahikkala, and T. Salakoski. Extracting complex biological events with rich graph-based feature sets. In

Proceedings of the Workshop on BioNLP: Shared Task, pages 10–18. Association for Computational Linguistics, 2009.

[51] C. Blaschke, M.A. Andrade, C. Ouzounis, and A. Valencia. Automatic extraction of biological information from scientific text: protein-protein interactions. In *Proc Int Conf Intell Syst Mol Biol*, volume 1999, pages 60–67, 1999.

[52] C. Blaschke and A. Valencia. The frame-based module of the Suiseki information extraction system. *IEEE Intelligent Systems*, pages 14–20, 2002.

[53] S. Bleuler, A. Prelic, and E. Zitzler. An EA framework for biclustering of gene expression data. In *Congress on Evolutionary Computation, 2004. CEC2004.*, volume 1, 2004.

[54] O. Bodenreider. The unified medical language system (UMLS): integrating biomedical terminology. *Nucleic Acids Research*, 32(Database Issue):D267, 2004.

[55] BM Bolstad, RA Irizarry, M. Astrand, and TP Speed. A comparison of normalization methods for high density oligonucleotide array data based on variance and bias. *Bioinformatics*, 19(2):185, 2003.

[56] E.I. Boyle, S. Weng, J. Gollub, H. Jin, D. Botstein, J.M. Cherry, and G. Sherlock. GO:: TermFinder–open source software for accessing Gene Ontology information and finding significantly enriched Gene Ontology terms associated with a list of genes. *Bioinformatics*, 20(18):3710, 2004.

[57] S. Brady and H. Shatkay. EpiLoc: a (working) text-based system for predicting protein subcellular location. In *Pac Symp Biocomput*, volume 604, page 15, 2008.

[58] K. Bryan, P. Cunningham, and N. Bolshakova. Application of Simulated Annealing to the Biclustering of Gene Expression Data. *IEEE Transactions on Information Technology in Biomedicine*, 10(3):519–525, 2006.

[59] R. Bunescu, R. Ge, R.J. Kate, E.M. Marcotte, R.J. Mooney, A.K. Ramani, and Y.W. Wong. Comparative experiments on learning information extractors for proteins and their interactions. *Artificial Intelligence in Medicine*, 33(2):139–155, 2005.

[60] C. Callison-Burch. Fast, Cheap, and Creative: Evaluating Translation Quality Using Amazon's Mechanical Turk. *Proceedings of EMNLP 2009*, 2009.

[61] C. Cano, T. Monaghan, A. Blanco, DP Wall, and L. Peshkin. Collaborative text-annotation resource for disease-centered relation extraction from biomedical text. *Journal of Biomedical Informatics*, 42(5):967–977, 2009.

[62] J.G. Caporaso, W.A. Baumgartner, D.A. Randolph, K.B. Cohen, and L. Hunter. MutationFinder: a high-performance system for extracting point mutation mentions from text. *Bioinformatics*, 23(14):1862, 2007.

[63] A. Carlson, J. Betteridge, R.C. Wang, E.R. Hruschka Jr, and T.M. Mitchell. Coupled Semi-Supervised Learning for Information Extraction. In *Proceedings of the Third ACM International Conference on Web Search and Data Mining (WSDM)*, page 110, 2010.

[64] B. Carpenter. Lazy Sparse Stochastic Gradient Descent for Regularized Multinomial Logistic Regression. Technical report, Alias-I, inc., 2008.

[65] B. Carpenter and B. Baldwin. Lingpipe, 2008. http://alias-i.com/lingpipe/.

[66] W.W. Chapman and K.B. Cohen. Current issues in biomedical text mining and natural language processing. *Journal of Biomedical Informatics*, 42(5):757–759, 2009.

[67] E. Charniak and M. Johnson. Coarse-to-fine n-best parsing and MaxEnt discriminative reranking. In *Proceedings of the 43rd Annual Meeting on Association for Computational Linguistics*, page 180. Association for Computational Linguistics, 2005.

[68] H. Chen and B.M. Sharp. Content-rich biological network constructed by mining PubMed abstracts. *BMC bioinformatics*, 5(1):147, 2004.

[69] Y. Cheng and G.M. Church. Biclustering of expression data. In *Proc Int Conf Intell Syst Mol Biol*, volume 8, pages 93–103, 2000. http://arep.med.harvard.edu/biclustering.

[70] H. Cho, I.S. Dhillon, Y. Guan, and S. Sra. Minimum sum-squared residue co-clustering of gene expression data. In *Proceedings of the fourth SIAM international conference on data mining*, pages 114–125, 2004.

[71] R.J. Cho, M.J. Campbell, E.A. Winzeler, L. Steinmetz, A. Conway, L. Wodicka, T.G. Wolfsberg, A.E. Gabrielian, D. Landsman, D.J. Lockhart, et al. A genome-wide transcriptional analysis of the mitotic cell cycle. *Molecular cell*, 2(1):65–73, 1998.

[72] S. Clancy. RNA splicing: introns, exons and spliceosome. *Nature Education*, 1(1), 2008.

[73] A.B. Clegg and A.J. Shepherd. Benchmarking natural-language parsers for biological applications using dependency graphs. *BMC bioinformatics*, 8(1):24, 2007.

[74] M. Collins and N. Duffy. Convolution kernels for natural language. *Advances in neural information processing systems*, 1:625–632, 2002.

[75] M. Colosimo, A. Morgan, A. Yeh, J. Colombe, and L. Hirschman. Data preparation and interannotator agreement: BioCreAtIvE task 1B. *BMC bioinformatics*, 6(Suppl 1):S12, 2005.

[76] P. Corbett and P. Murray-Rust. High-throughput identification of chemistry in life science texts. *Lecture Notes in Computer Science*, 4216:107, 2006.

[77] D.P.A. Corney, B.F. Buxton, W.B. Langdon, and D.T. Jones. BioRAT: extracting biological information from full-length papers. *Bioinformatics*, 20(17):3206–3213, 2004.

[78] M. Craven and J. Kumlien. Constructing biological knowledge bases by extracting information from text sources. In *Proceedings of the Seventh International Conference on Intelligent Systems for Molecular Biology*, pages 77–86. Heidelberg, Germany, 1999. http://www.biostat.wisc.edu/ craven/ie/.

[79] D.H. Cunningham, D.D. Maynard, D.K. Bontcheva, and M.V. Tablan. GATE: A framework and graphical development environment for robust NLP tools and applications. In *Proceedings of the 40th Anniversary Meeting of the Association for Computational Linguistics*, 2002. www.gate.ack.uk .

[80] N. Daraselia, A. Yuryev, S. Egorov, S.Ñovichkova, A.Ñikitin, and I. Mazo. Extracting human protein interactions from MEDLINE using a full-sentence parser. *Bioinformatics*, 20(5):604, 2004.

[81] J.S. De Bonet, C.L. Isbell, and P. Viola. MIMIC: Finding optima by estimating probability densities. *Advances in neural information processing systems*, pages 424–430, 1997.

[82] F. De Smet, J. Mathys, K. Marchal, G. Thijs, B. De Moor, and Y. Moreau. Adaptive quality-based clustering of gene expression profiles. *Bioinformatics*, 18(5):735, 2002.

[83] K. Deb. *Multi-objective optimization using evolutionary algorithms*. Wiley, 2001.

[84] K. Deb, A. Pratap, S. Agarwal, T. Meyarivan, A. Fast, and E.M.G. Algorithm. NSGA-II. *IEEE transactions on evolutionary computation*, 6(2), 2002.

[85] K. Degtyarenko, P. de Matos, M. Ennis, J. Hastings, M. Zbinden, A. McNaught, R. Alcantara, M. Darsow, M. Guedj, and M. Ashburner. ChEBI: a database and ontology for chemical entities of biological interest. *Nucleic acids research*, 36(Database issue):D344, 2008.

[86] D. Dembele and P. Kastner. Fuzzy C-means method for clustering microarray data. *Bioinformatics*, 19(8):973, 2003.

[87] K. Denecke. Semantic structuring of and information extraction from medical documents using the UMLS. *Methods of information in medicine*, 47(5):425, 2008.

[88] J.L. DeRisi, V.R. Iyer, and P.O. Brown. Exploring the metabolic and genetic control of gene expression on a genomic scale. *Science*, 278(5338):680, 1997.

[89] I.S. Dhillon. Co-clustering documents and words using bipartite spectral graph partitioning. In *Proceedings of the seventh ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 269–274. ACM New York, NY, USA, 2001.

[90] V. Di Gesú, R. Giancarlo, L. Bosco, A. Raimondi, and D. Scaturro. GenClust: A genetic algorithm for clustering gene expression data. *BMC bioinformatics*, 6(1):289, 2005.

[91] E.P. Diamandis and D.E. van der Merwe. Plasma protein profiling by mass spectrometry for cancer diagnosis: opportunities and limitations. *Clinical Cancer Research*, 11(3):963, 2005.

[92] J. Ding, D. Berleant, D.Ñettleton, and E. Wurtele. Mining MEDLINE: Abstracts, sentences, or phrases? In *Pacific Symposium on Biocomputing 2002: Kauai, Hawaii, 3-7 January 2002*, page 326. World Scientific Publishing Company, 2002.

[93] J. Ding, D. Berleant, J. Xu, and AW Fulmer. Extracting biochemical interactions from MEDLINE using a link grammar parser. In *15th IEEE International Conference on Tools with Artificial Intelligence, 2003. Proceedings*, pages 467–471, 2003.

[94] F. Divina and J.S. Aguilar-Ruiz. Biclustering of expression data with evolutionary computation. *IEEE transactions on knowledge and data engineering*, pages 590–602, 2006.

[95] F. Divina and J.S. Aguilar-Ruiz. A multi-objective approach to discover biclusters in microarray data. In *Proceedings of the 9th annual conference on Genetic and evolutionary computation*, page 392. ACM, 2007.

[96] K. Do, R.Ñikolova, P. Roebuck, and B. Broom. Software geneclust, 2002. http://odin.mdacc.tmc.edu/ kim/geneclust/.

[97] P. Donmez, J.G. Carbonell, and J. Schneider. Efficiently learning the accuracy of labeling sources for selective sampling. In *Proceedings of the 15th ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 259–268. ACM New York, NY, USA, 2009.

[98] J. Dopazo. Functional interpretation of microarray experiments. *OMICS: A Journal of Integrative Biology*, 10(3):398–410, 2006.

[99] S. Douglas, G. Montelione, and M. Gerstein. PubNet: a flexible system for visualizing literature derived networks. *Genome biology*, 6(9):R80, 2005.

[100] R.D. Dowell, R.M. Jokerst, A. Day, S.R. Eddy, and L. Stein. The distributed annotation system. *BMC bioinformatics*, 2(1):7, 2001.

[101] Z. Du and F. Lin. A novel parallelization approach for hierarchical clustering. *Parallel Computing*, 31(5):523–527, 2005.

[102] S. Dudoit and JYH. Yang. *Bioconductor R Packages for Exploratory Analysis and Normalization of cDNA Microarray Data*. Ed. Springer London., 2003. The Analysis of Gene Expression Data, Methods and Software.

[103] M.B. Eisen, P.T. Spellman, P.O. Brown, and D. Botstein. Cluster analysis and display of genome-wide expression patterns. *Proceedings of the National Academy of Sciences*, 95(25):14863, 1998.

[104] G. Erkan, A. Ozgur, and D.R. Radev. Semi-supervised classification for extracting protein interaction sentences using dependency parsing. In *Proceedings of the 2007 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning (EMNLP-CoNLL)*, volume 1, pages 228–237, 2007.

[105] M. Fiedler. Algebraic connectivity of graphs. *Czechoslovak Mathematical Journal*, 23(2):298–305, 1973.

[106] J. Finkel, S. Dingare, C. Manning, M.Ñissim, B. Alex, and C. Grover. Exploring the boundaries: Gene and protein identification in biomedical text. *BMC bioinformatics*, 6(Suppl 1):S5, 2005.

[107] G. Forman. An extensive empirical study of feature selection metrics for text classification. *The Journal of Machine Learning Research*, 3:1289–1305, 2003.

[108] W.J. Frawley, G. Piatetsky-Shapiro, and C.J. Matheus. Knowledge discovery in databases: An overview. *Ai Magazine*, 13(3):57–70, 1992.

[109] C. Friedman, P. Kra, H. Yu, M. Krauthammer, and A. Rzhetsky. GENIES: a natural-language processing system for the extraction of molecular pathways from journal articles. *BIOINFORMATICS-OXFORD-*, 17:74–82, 2001.

[110] C. Friedman, L. Shagina, Y. Lussier, and G. Hripcsak. Automated encoding of clinical documents based on natural language processing. *Journal of the American Medical Informatics Association*, 11(5):392–402, 2004.

[111] L. Fu and E. Medico. FLAME, a novel fuzzy clustering method for the analysis of DNA microarray data. *BMC bioinformatics*, 8(1):3, 2007.

[112] K. Fukuda, A. Tamura, T. Tsunoda, and T. Takagi. Toward information extraction: identifying protein names from biological papers. In *Pacific Symposium on Biocomputing. Pacific Symposium on Biocomputing*, page 707. Pac Symp Biocomput, 1998.

[113] K. Fundel, R. Kuffner, and R. Zimmer. RelEx–relation extraction using dependency parse trees. *Bioinformatics*, 23(3):365, 2007.

[114] X. Gan, A.W.C. Liew, and H. Yan. Discovering biclusters in gene expression data based on high-dimensional linear geometries. *BMC bioinformatics*, 9(1):209, 2008.

[115] A.P. Gasch and M.B. Eisen. Exploring the conditional coregulation of yeast gene expression through fuzzy k-means clustering. *Genome Biol*, 3(11):1–22, 2002.

[116] I. Gath and A.B. Geva. Unsupervised optimal fuzzy clustering. *IEEE Transactions on pattern analysis and machine intelligence*, 11(7):773–780, 1989.

[117] A. Genkin, D.D. Lewis, and D. Madigan. BMR: Bayesian Multinomial Regression Software. *DIMACS, http://www. stat. rutgers. edu/madigan/BMR/, accessed on*, 14(02), 2008.

[118] G. Getz, E. Levine, and E. Domany. Coupled two-way clustering analysis of gene microarray data. *Proceedings of the National Academy of Sciences*, 97(22):12079, 2000.

[119] A. Ghouila, S.B. Yahia, D. Malouche, H. Jmel, D. Laouini, F.Z. Guerfali, and S. Abdelhak. Application of Multi-SOM clustering approach to macrophage gene expression analysis. *Infection, Genetics and Evolution*, 9(3):328–336, 2009.

[120] F.D. Gibbons and F.P. Roth. Judging the quality of gene expression-based clustering methods using gene annotation. *Genome Research*, 12(10):1574, 2002.

[121] F. Giorgini and P.J. Muchowski. Connecting the dots in Huntington's disease with protein interaction networks. *Genome Biol*, 6(3), 2005.

[122] A. Gitter, C. Baral, and G. Gonzalez. Biomedical information extraction through deep parsing and syntactic role matching, 2007. http://www.public.asu.edu/ ajgitter/acm/.

[123] G. Glass. Marginalia Web Annotation Project. http://www.geof.net/code/annotation.

[124] G.H. Golub and C.F. Van Loan. *Matrix computations*. Johns Hopkins Univ Pr, 1996.

[125] TR Golub, DK Slonim, P. Tamayo, C. Huard, M. Gaasenbeek, JP Mesirov, H. Coller, ML Loh, JR Downing, MA Caligiuri, et al. Molecular classification of cancer: class discovery and class prediction by gene expression monitoring. *Science*, 286(5439):531, 1999.

[126] J.T. Goodman. Exponential priors for maximum entropy models, jan, 27 2009. US Patent 7,483,813.

[127] Z. GuoDong and S. Jian. Exploring deep knowledge resources in biomedical name recognition. In *Proceedings of the International Joint Workshop on Natural Language Processing in Biomedicine and its Applications*, pages 96–99. Association for Computational Linguistics, 2004.

[128] L. Hagen and A.B. Kahng. New spectral methods for ratio cut partitioning and clustering. *IEEE Trans. Computer-Aided Design*, 11(9):1074–1085, 1992.

[129] J. Hakenberg, C. Plake, U. Leser, H. Kirsch, and D. Rebholz-Schuhmann. LLL'05 challenge: Genic interaction extraction-identification of language patterns based on alignment and finite state automata.

[130] J. Hakenberg, C. Plake, L. Royer, H. Strobelt, U. Leser, and M. Schroeder. Gene mention normalization and interaction extraction with context models and sentence motifs. *Genome Biology*, 9(Suppl 2):S14, 2008.

[131] DJ Hand, H. Mannila, and P. Smyth. *Principles of data mining*. The MIT Press, 2001.

[132] Y. Hao, X. Zhu, M. Huang, and M. Li. Discovering patterns to extract protein-protein interactions from the literature: Part II. *Bioinformatics*, 21(15):3294, 2005.

[133] JA Hartigan. Direct clustering of a data matrix. *Journal of the American Statistical Association*, pages 123–129, 1972.

[134] T. Hastie, R. Tibshirani, M. Eisen, P. Brown, D. Ross, U. Scherf, J. Weinstein, A. Alizadeh, L. Staudt, and D. Botstein. Gene shaving : A new class of clustering methods for expression arrays. Technical report, Department of Statistics, Stanford University., 2000.

[135] T. Hastie, R. Tibshirani, M.B. Eisen, A. Alizadeh, R. Levy, L. Staudt, W.C. Chan, D. Botstein, and P. Brown. Gene shaving' as a method for identifying distinct sets of genes with similar expression patterns. *Genome Biology*, 1(2):1–0003, 2000.

[136] M. A. Hearst. What is text mining?, 2003. http://www.ischool.berkeley.edu/h̃earst/text-mining-html.

[137] M.A. Hearst. Untangling text data mining. In *Proceedings of the 37th annual meeting of the Association for Computational Linguistics on Computational Linguistics*, pages 3–10. Association for Computational Linguistics Morristown, NJ, USA, 1999.

[138] J. Hernández Orallo, MJ Ramírez Quintana, and C. Ferri Ramírez. *Introducción a la Minería de Datos*. España, Madrid: Pearson educacion SA, 2004.

[139] J. Herrero, A. Valencia, and J. Dopazo. A hierarchical unsupervised growing neural network for clustering gene expression patterns. *Bioinformatics*, 17(2):126, 2001.

[140] J.N. Hirschhorn and M.J. Daly. Genome-wide association studies for common diseases and complex traits. *Nature Reviews Genetics*, 6(2):95–108, 2005.

[141] L. Hirschman, A. Yeh, C. Blaschke, and A. Valencia. Overview of BioCreAtIvE: critical assessment of information extraction for biology. *BMC bioinformatics*, 6(Suppl 1):S1, 2005.

[142] R. Hoffmann and A. Valencia. A gene network for navigating the literature. *Phys. Rev. E Stat. Nonlin. Soft Matter Phys*, 65:065102, 2002.

[143] J.H. Holland. *Adaptation in natural and artificial systems*. MIT press Cambridge, MA, 1992.

[144] D.A. Hosack, G. Dennis Jr, B.T. Sherman, H.C. Lane, and R.A. Lempicki. Identifying biological themes within lists of genes with EASE. *Genome Biol*, 4(10):R70, 2003.

[145] M. Huang, X. Zhu, Y. Hao, D.G. Payan, K. Qu, and M. Li. Discovering patterns to extract protein-protein interactions from full texts. *Bioinformatics-Oxford*, 20(18):3604–3612, 2004.

[146] M. Huang, X. Zhu, Y. Hao, D.G. Payan, K. Qu, and M. Li. Discovering patterns to extract protein-protein interactions from full texts. *Bioinformatics-Oxford*, 20(18):3604–3612, 2004. http://spies.cs.tsinghua.edu.cn.

[147] M. Hubert and S. Engelen. Robust PCA and classification in biosciences. *Bioinformatics*, 20(11):1728, 2004.

[148] S. Humbert and F. Saudou. The ataxia-ome: connecting disease proteins of the cerebellum. *Cell*, 125(4):645–647, 2006.

[149] L. Hunter and K.B. Cohen. Biomedical language processing: what's beyond PubMed? *Molecular Cell*, 21(5):589–594, 2006.

[150] JW Huss, C. Orozco, J. Goodale, C. Wu, S. Batalov, T.J. Vickers, F. Valafar, and A.I. Su. A gene wiki for community annotation of gene function. *PLoS Biol*, 6(7):e175, 2008.

[151] T. Ideker and R. Sharan. Protein networks in disease. *Genome Research*, 18(4):644, 2008.

[152] I. Inza, B. Calvo, R. Armañanzas, E. Bengoetxea, P. Larrañaga, and JA. Lozano. Machine learning: an indispensable tool in bioinformatics. *Methods in Molecular Biology*, 593:25–48, 2010.

[153] P. Jackson and I. Moulinier. *Natural language processing for online applications: Text retrieval, extraction and categorization*. John Benjamins Publishing Co, 2007.

[154] H. Jang, J. Lim, J.H. Lim, S.J. Park, K.C. Lee, and S.H. Park. Finding the evidence for protein-protein interactions from PubMed abstracts. *Bioinformatics*, 22(14):e220, 2006.

[155] R. Jelier, G. Jenster, LCJ Dorssers, CC Van Der Eijk, EM Van Mulligen, B. Mons, and JA Kors. Co-occurrence based meta-analysis of scientific texts: retrieving biological relationships between genes. *Bioinformatics*, 21(9):2049, 2005.

[156] L.J. Jensen, J. Saric, P. Bork, et al. Literature mining for the biologist: from information retrieval to biological discovery. *Nature Reviews Genetics*, 7(2):119–129, 2006.

[157] T.K. Jenssen, A. Lægreid, J. Komorowski, and E. Hovig. A literature network of human genes for high-throughput analysis of gene expression. *Nature Genetics*, 28(1):21–28, 2001.

[158] A. Jentzsch, O. Hassanzadeh, C. Bizer, B. Andersson, and S. Stephens. Enabling Tailored Therapeutics with Linked Data. In *Proceedings of the 2nd Workshop about Linked Data on the Web*, 2009.

[159] A. Jentzsch, J. Zhao, O. Hassanzadeh, K.H. Cheung, M. Samwald, and B. Andersson. Linking Open Drug Data.
http://triplify.org/files/challenge_2009/LODD.pdf.

[160] D. Jiang, C. Tang, and A. Zhang. Cluster analysis for gene expression data: A survey. *IEEE Transactions on Knowledge and Data Engineering*, pages 1370–1386, 2004.

[161] H.L. Johnson, W.A. Baumgartner Jr, M. Krallinger, K.B. Cohen, and L. Hunter. Refactoring corpora. In *Proceedings of the BioNLP Workshop on Linking Natural Language Processing and Biology at HLT-NAACL*, volume 6, pages 116–117, 2006. http://bionlp-corpora.sourceforge.net/picorpus/index.shtml.

[162] A. Joshi, Y. Van de Peer, and T. Michoel. Analysis of a Gibbs sampler method for model-based clustering of gene expression data. *Bioinformatics*, 24(2):176, 2008.

[163] A.S. Juncker, L.J. Jensen, A. Pierleoni, A. Bernsel, M.L. Tress, P. Bork, G. Von Heijne, A. Valencia, C.A. Ouzounis, R. Casadio, et al. Sequence-based feature prediction and annotation of proteins. *Genome Biology*, 10(2):206, 2009.

[164] R. Kabiljo, A.B. Clegg, and A.J. Shepherd. A realistic assessment of methods for extracting gene/protein interactions from free text. *BMC bioinformatics*, 10(1):233, 2009.

[165] M. Kanehisa, S. Goto, S. Kawashima, Y. Okuno, and M. Hattori. The KEGG resource for deciphering the genome. *Nucleic acids research*, 32(Database Issue):D277, 2004.

[166] MG Kann. Protein interactions and disease: computational approaches to uncover the etiology of diseases. *Briefings in bioinformatics*, 8(5):333, 2007.

[167] Y. Kano, W.A. Baumgartner, L. McCrohon, S. Ananiadou, K.B. Cohen, L. Hunter, and J. Tsujii. U-Compare: share and compare text mining tools with UIMA. *Bioinformatics*, 25(15):1997, 2009.

[168] J. Kennedy, R.C. Eberhart, et al. Particle swarm optimization. In *Proceedings of IEEE international conference on neural networks*, volume 4, pages 1942–1948. Piscataway, NJ: IEEE, 1995.

[169] G. Kerr, HJ Ruskin, M. Crane, and P. Doolan. Techniques for clustering gene expression data. *Computers in biology and medicine*, 38(3):283, 2008.

[170] H. Kim, G.H. Golub, and H. Park. Missing value estimation for DNA microarray gene expression data: local least squares imputation. *Bioinformatics*, 21(2):187, 2005.

[171] J. Kim and J.C. Park. BioIE: retargetable information extraction and ontological annotation of biological interactions from the literature. *Journal of bioinformatics and computational biology*, 2(3):551–568, 2004. http://bioie.biopathway.org/.

[172] J. Kim, Z. Zhang, J.C. Park, and S.K. Ng. BioContrasts: extracting and exploiting protein-protein contrastive relations from biomedical literature. *Bioinformatics*, 22(5):597, 2006. http://biocontrasts.i2r.a-star.edu.sg/BioContrasts-testcorpus.html.

[173] J.D. Kim, T. Ohta, Y. Tateisi, and J. Tsujii. GENIA corpus-a semantically annotated corpus for bio-textmining. *Bioinformatics-Oxford*, 19(1):180–182, 2003.

[174] J.D. Kim, T. Ohta, J. Tsujii, et al. Corpus annotation for mining biomedical events from literature. *BMC bioinformatics*, 9(1):10, 2008. http://www-tsujii.is.s.u-tokyo.ac.jp/GENIA/home/wiki.cgi.

[175] J.D. Kim, T. Ohta, Y. Tsuruoka, Y. Tateisi, and N. Collier. Introduction to the bio-entity recognition task at JNLPBA. In *Proceedings of the International Joint Workshop on Natural Language Processing in Biomedicine and its Applications*, pages 70–75. Association for Computational Linguistics, 2004.

[176] S. Kim, S.Y. Shin, I.H. Lee, S.J. Kim, R. Sriram, and B.T. Zhang. PIE: an online prediction system for protein–protein interactions from text. *Nucleic Acids Research*, 36(Web Server issue):W411, 2008.

[177] S. Kim, J. Yoon, and J. Yang. Kernel approaches for genic interaction extraction. *Bioinformatics*, 24(1):118, 2008.

[178] S.Y. Kim, J.W. Lee, and J.S. Bae. Effect of data normalization on fuzzy clustering of DNA microarray data. *BMC bioinformatics*, 7(1):134, 2006.

[179] S. Kirkpatric, CD Gelatt, and MP Vecchi. Optimization by simulated annealing. *Science*, 220(4598):671–680, 1983.

[180] H. Kirsch, S. Gaudan, and D. Rebholz-Schuhmann. Distributed modules for text annotation and IE applied to the biomedical domain. *International journal of medical informatics*, 75(6):496–500, 2006.

[181] D. Klein and C.D. Manning. Accurate unlexicalized parsing. In *Proceedings of the 41st Annual Meeting on Association for Computational Linguistics-Volume 1*, pages 423–430. Association for Computational Linguistics Morristown, NJ, USA, 2003.

[182] D. Klein and C.D. Manning. Fast exact inference with a factored model for natural language parsing. *Advances in Neural Information Processing Systems*, pages 3–10, 2003.

[183] W. Klimke, R. Agarwala, A. Badretdin, S. Chetvernin, S. Ciufo, B. Fedorov, B. Kiryutin, K. O'Neill, W. Resch, S. Resenchuk, et al. The National Center for Biotechnology Information's Protein Clusters Database. *Nucleic acids research*, 37(Database issue):D216, 2009.

[184] J.D. Knowles and D.W. Corne. Approximating the nondominated front using the pareto archived evolution strategy. *Evolutionary computation*, 8(2):149–172, 2000.

[185] I. Kononenko and M. Kukar. *Machine learning and data mining: introduction to principles and algorithms*. Horwood Pub Ltd, 2007.

[186] M. Krallinger, A. Morgan, L. Smith, F. Leitner, L. Tanabe, J. Wilbur, L. Hirschman, and A. Valencia. Evaluation of text-mining systems for biology: overview of the Second BioCreative community challenge. *Genome biology*, 9(Suppl 2):S1, 2008.

[187] M. Krallinger, A. Valencia, and L. Hirschman. Linking genes to literature: text mining, information extraction, and retrieval applications for biology. *Genome biology*, 9(Suppl 2):S8, 2008.

[188] R. Krishnapuram and J.M. Keller. A possibilistic approach to clustering. *IEEE transactions on fuzzy systems*, 1(2):98–110, 1993.

[189] M. Kull and J. Vilo. Fast approximate hierarchical clustering using similarity heuristics. *BioData Mining*, 1:9, 2008.

[190] P. Larrañaga, B. Calvo, R. Santana, C. Bielza, J. Galdiano, I. Inza, JA Lozano, R. Armañanzas, G. Santafé, A. Pérez, et al. Machine learning in bioinformatics. *Briefings in bioinformatics*, 7(1):86, 2006.

[191] P. Larranaga and J.A. Lozano. *Estimation of distribution algorithms: A new tool for evolutionary computation*. Kluwer Academic Pub, 2002.

[192] P. Larranaga, JA Lozano, and E. Bengoetxea. Estimation of distribution algorithms based on multivariate normal and Gaussian networks. Technical report, University of the Basque Country, Technical Report: KZZA-IK-1-01, 2001.

[193] P. Larrañaga, R Etxeberria, J.A. Lozano, and J. M. Peña. Combinatorial optimization by learning and simulation of bayesian netwoks. In *Proceedings of the 16th Conference on Uncertainty in Artificial Intelligence.*, pages 343–352, 2000.

[194] M.T. Laub, H.H. McAdams, T. Feldblyum, C.M. Fraser, and L. Shapiro. Global analysis of the genetic network controlling a bacterial cell cycle. *Science*, 290(5499):2144, 2000.

[195] L. Lazzeroni and A. Owen. Plaid models for gene expression data. *Statistica Sinica*, 12(1):61–86, 2002.

[196] M. Lease and E. Charniak. Parsing Biomedical Literature. In *Natural Language Processing – IJCNLP 2005*, pages 58–69, 2005. http://bllip.cs.brown.edu/resources.shtml.

[197] C.H. Lee, C.H. Wu, and H.C. Yang. Text Mining of Clinical Records for Cancer Diagnosis. In *Proceedings of the Second International Conference on Innovative Computing, Informatio and Control*, page 172. IEEE Computer Society, 2007.

[198] F. Leitner, M. Krallinger, C. Rodriguez-Penagos, J. Hakenberg, C. Plake, C.J. Kuo, C.N. Hsu, R. Tsai, H.C. Hung, W. Lau, et al. Introducing meta-services for biomedical information extraction. *Genome Biology*, 9(Suppl 2):S6, 2008.

[199] U. Leser and J. Hakenberg. What makes a gene name? Named entity recognition in the biomedical literature. *Briefings in Bioinformatics*, 6(4):357, 2005.

[200] C.S. Leslie, E. Eskin, A. Cohen, J. Weston, and W.S. Noble. Mismatch string kernels for discriminative protein classification. *Bioinformatics*, 20(4):467–476, 2004.

[201] J. Lim, T. Hao, C. Shaw, A.J. Patel, G. Szabó, J.F. Rual, C.J. Fisk, N. Li, A. Smolyar, D.E. Hill, et al. A protein–protein interaction network for human inherited ataxias and disorders of Purkinje cell degeneration. *Cell*, 125(4):801–814, 2006.

[202] J. Liu, Z. Li, X. Hu, and Y. Chen. Biclustering of microarray data with MOSPO based on crowding distance. *BMC bioinformatics*, 10(Suppl 4):S9, 2009.

[203] X. Liu and L. Wang. Computing the maximum similarity bi-clusters of gene expression data. *Bioinformatics*, 23(1):50, 2007.

[204] W. Long. Lessons extracting diseases from discharge summaries. In *AMIA Annual Symposium Proceedings*, volume 2007, page 478. American Medical Informatics Association, 2007.

[205] Y. Lu, S. Lu, F. Fotouhi, Y. Deng, and S.J. Brown. Incremental genetic k-means algorithm and its application in gene expression data analysis. *BMC bioinformatics*, 5(1):172, 2004.

[206] F. Luo, L. Khan, F. Bastani, I.L. Yen, and J. Zhou. A dynamically growing self-organizing tree (DGSOT) for hierarchical clustering gene expression profiles. *Bioinformatics*, 20(16):2605–2617, 2004.

[207] López M., Mallorquín P., and M Vega. Microarrays y biochips de adn, 2002.

[208] López M., Mallorquín P., and M Vega. Aplicaciones de los microarrays y los biochips en salud humana., 2005. http://www.genes.org/02_cono/02_cono.cfm?pag=0310#8.

[209] S. Ma and J. Huang. Penalized feature selection and classification in bioinformatics. *Briefings in Bioinformatics*, 2008.

[210] S.C. Madeira and A.L. Oliveira. Biclustering algorithms for biological data analysis: a survey. *IEEE Transactions on computational Biology and Bioinformatics*, pages 24–45, 2004.

[211] H. Maier, S. Dohr, K. Grote, S. O'Keeffe, T. Werner, M.H. de Angelis, and R. Schneider. LitMiner and WikiGene: identifying problem-related key players of gene regulation using publication abstracts. *Nucleic acids research*, 33(Web Server Issue):W779, 2005.

[212] E.R. Mardis. The impact of next-generation sequencing technology on genetics. *Trends in Genetics*, 24(3):133–141, 2008.

[213] A. Mateos, J. Dopazo, R. Jansen, Y. Tu, M. Gerstein, and G. Stolovitzky. Systematic learning of gene functional classes from DNA array expression data by using multilayer perceptrons. *Genome Research*, 12(11):1703, 2002.

[214] S. Mathivanan, B. Periaswamy, TKB Gandhi, K. Kandasamy, S. Suresh, R. Mohmood, YL Ramachandra, and A. Pandey. An evaluation of human protein-protein interaction data in the public domain. *BMC bioinformatics*, 7(Suppl 5):S19, 2006.

[215] U. Maulik and A. Mukhopadhyay. Simulated annealing based automatic fuzzy clustering combined with ANN classification for analyzing microarray data. *Computers and Operations Research*, 2009.

[216] U. Maulik, A. Mukhopadhyay, and S. Bandyopadhyay. Combining Pareto-optimal clusters using supervised learning for identifying co-expressed genes. *BMC bioinformatics*, 10(1):27, 2009.

[217] A.T. McCray, A.C. Browne, and O. Bodenreider. The lexical properties of the gene ontology. In *Proceedings of the AMIA Symposium*, page 504. American Medical Informatics Association, 2002.

[218] R. McDonald and F. Pereira. Identifying gene and protein mentions in text using conditional random fields. *BMC bioinformatics*, 6(Suppl 1):S6, 2005.

[219] H. Mcwilliam, F. Valentin, M. Goujon, W. Li, M.Ñarayanasamy, J. Martin, T. Miyar, and R. Lopez. Web services at the European Bioinformatics Institute-2009. *Nucleic Acids Research*, 37(Web Server issue):W6, 2009.

[220] I. Medina, D. Montaner, J. Tarraga, and J. Dopazo. Prophet, a web-based tool for class prediction using microarray data. *Bioinformatics*, 23(3):390, 2007.

[221] E. Mejia-Roa, P. Carmona-Saez, R.Ñogales, C. Vicente, M. Vazquez, XY Yang, C. Garcia, F. Tirado, and A. Pascual-Montano. bioNMF: a web-based tool for nonnegative matrix factorization in biology. *Nucleic Acids Research*, 36(Web Server issue):W523, 2008.

[222] G.W. Milligan and M.C. Cooper. An examination of procedures for determining the number of clusters in a data set. *Psychometrika*, 50(2):159–179, 1985.

[223] P. Minguez, S. Gotz, D. Montaner, F. Al-Shahrour, and J. Dopazo. SNOW, a web-based tool for the statistical analysis of protein-protein interaction networks. *Nucleic Acids Research*, 37:W109–W114, 2009.

[224] T.M. Mitchell. *Machine learning*. McGraw-Hill, 1997.

[225] S. Mitra and H. Banka. Multi-objective evolutionary biclustering of gene expression data. *Pattern Recognition*, 39(12):2464–2477, 2006.

[226] T. Mitsumori, S. Fation, M. Murata, K. Doi, and H. Doi. Gene/protein name recognition based on support vector machine using dictionary as features. *BMC bioinformatics*, 6(Suppl 1):S8, 2005.

[227] M. Miwa, R. Saetre, Y. Miyao, and J. Tsujii. Protein-protein interaction extraction by leveraging multiple kernels and parsers. *International journal of medical informatics*, 2009.

[228] Y. Miyao, K. Sagae, R. Saetre, T. Matsuzaki, and J. Tsujii. Evaluating contributions of natural language parsers to protein-protein interaction extraction. *Bioinformatics*, 25(3):394, 2009.

[229] B. Mons, M. Ashburner, C. Chichester, E. van Mulligen, M. Weeber, J. den Dunnen, G.J. van Ommen, M. Musen, M. Cockerill, H. Hermjakob, et al. Calling on a million minds for community annotation in WikiProteins. *Genome biology*, 9(5):R89, 2008.

[230] D. Montaner, P. Minguez, F. Al-Shahrour, and J. Dopazo. Gene set internal coherence in the context of functional profiling. *BMC genomics*, 10(1):197, 2009.

[231] U. Mudunuri, R. Stephens, D. Bruining, D. Liu, and F.J. Lebeda. botXminer: mining biomedical literature with a new web-based application. *Nucleic acids research*, 34(Web Server issue):W748, 2006.

[232] H. Muhlenbein. The equation for response to selection and its use for prediction. *Evolutionary Computation*, 5(3):303–346, 1997.

[233] A. Mukhopadhyay and U. Maulik. Towards improving fuzzy clustering using support vector machine: Application to gene expression data. *Pattern Recognition*, 42(11):2744–2763, 2009.

[234] I.Ñeamatullah, M.M. Douglass, L.H. Lehman, A. Reisner, M. Villarroel, W.J. Long, P. Szolovits, G.B. Moody, R.G. Mark, and G.D. Clifford. Automated de-identification of free-text medical records. *BMC Medical Informatics and Decision Making*, 8(1):32, 2008.

[235] A.Y. Ng, M.I. Jordan, and Y. Weiss. On spectral clustering: Analysis and an algorithm. *Advances in neural information processing systems*, 2:849–856, 2002.

[236] C.Ñobata, N. Collier, and J. Tsujii. Automatic term identification and classification in biology texts. In *Proc. of the 5th NLPRS*, pages 369–374, 1999.

[237] G.Ñowak and R. Tibshirani. Complementary hierarchical clustering. *Biostatistics*, 9(3):467, 2008.

[238] T. Ono, H. Hishigaki, A. Tanigami, and T. Takagi. Automated extraction of information on protein-protein interactions from the biological literature. *Bioinformatics*, 17(2):155, 2001.

[239] S. Pakhomov, S. Bjornsen, P. Hanson, and S. Smith. Quality Performance Measurement Using the Text of Electronic Medical Records. *Medical Decision Making*, 28(4):462, 2008.

[240] J.C. Park, H.S. Kim, and J.J. Kim. Bidirectional incremental parsing for automatic pathway identification with combinatory categorial grammar. In *Pac Symp Biocomput*, volume 2001, pages 396–407, 2001.

[241] RD Pascual-Marqui, AD Pascual-Montano, K. Kochi, and JM Carazo. Smoothly distributed fuzzy c-means: a new self-organizing map. *Pattern Recognition*, 34(12):2395–2402, 2001.

[242] A. Pascual-Montano, P. Carmona-Saez, M. Chagoyen, F. Tirado, J.M. Carazo, and R.D. Pascual-Marqui. bioNMF: a versatile tool for non-negative matrix factorization in biology. *BMC bioinformatics*, 7(1):366, 2006.

[243] T.A. Patterson, E.K. Lobenhofer, S.B. Fulmer-Smentek, P.J. Collins, T.M. Chu, W. Bao, H. Fang, E.S. Kawasaki, J. Hager, I.R. Tikhonova, et al. Performance comparison of one-color and two-color platforms within the Microarray Quality Control(MAQC) project. *Nature Biotechnology*, 24(9):1140–1150, 2006.

[244] T.A. Pearson and T.A. Manolio. How to interpret a genome-wide association study. *Jama*, 299(11):1335, 2008.

[245] R. Peeters. The maximum edge biclique problem is NP-complete. *Discrete Applied Mathematics*, 131(3):651–654, 2003.

[246] M. Pelikan, D.E. Goldberg, and E. Cantu-Paz. BOA: The Bayesian optimization algorithm. In *Proceedings of the Genetic and Evolutionary Computation Conference GECCO-99*, volume 1, pages 525–532, 1999.

[247] J.P. Pestian, C. Brew, P. Matykiewicz, DJ Hovermale, N. Johnson, K.B. Cohen, and W. Duch. A shared task involving multi-label classification of clinical free text. In *Proceedings of the Workshop on BioNLP 2007: Biological, Translational, and Clinical Language Processing*, pages 97–104. Association for Computational Linguistics, 2007.

[248] T.M. Phuong, D. Lee, and K.H. Lee. Learning rules to extract protein interactions from biomedical text. *Lecture notes in computer science*, pages 148–158, 2003.

[249] C. Plake, T. Schiemann, M. Pankalla, J. Hakenberg, and U. Leser. AliBaba: PubMed as a graph. *Bioinformatics*, 22(19):2444, 2006.

[250] N.L.M.M. Pochet, F.A.L. Janssens, F. De Smet, K. Marchal, J.A.K. Suykens, and B.L.R. De Moor. MACBETH: a microarray classification benchmarking tool. *Bioinformatics*, 21(14):3185, 2005.

[251] A. Prelic, S. Bleuler, P. Zimmermann, A. Wille, P. Buhlmann, W. Gruissem, L. Hennig, L. Thiele, and E. Zitzler. A systematic comparison and evaluation

of biclustering methods for gene expression data. *Bioinformatics*, 22(9):1122, 2006.

[252] HU Prokosch and T. Ganslandt. Perspectives for Medical Informatics: Reusing the Electronic Medical Record for Clinical Research. *Methods Inf Med*, 48(1):38–44, 2009.

[253] J. Pustejovsky, J. Castano, J. Zhang, M. Kotecki, and B. Cochran. Robust relational parsing over biomedical literature: Extracting inhibit relations. In *Pacific Symposium on Biocomputing 2002: Kauai, Hawaii, 3-7 January 2002*, page 362. World Scientific Publishing Company, 2002.

[254] S. Pyysalo. *A Dependency Parsing Approach to Biomedical Text Mining*. PhD thesis, Turku Center for Computer Science, 2008.

[255] S. Pyysalo, F. Ginter, J. Heimonen, J. Bjorne, J. Boberg, J. Jarvinen, and T. Salakoski. BioInfer: A corpus for information extraction in the biomedical domain. *BMC bioinformatics*, 8(1):50, 2007. `http://www.it.utu.fi/BioInfer`.

[256] Y. Qu and S. Xu. Supervised cluster analysis for microarray data based on multivariate Gaussian mixture. *Bioinformatics*, 20(12):1905–1913, 2004.

[257] J. Quackenbush. Microarray data normalization and transformation. *nature genetics*, 32(supp):496–501, 2002.

[258] A. Ramani, R. Bunescu, R. Mooney, and E. Marcotte. Consolidating the set of known human protein-protein interactions in preparation for large-scale mapping of the human interactome. *Genome Biology*, 6(5):R40, 2005.

[259] V.C. Raykar, S. Yu, L.H. Zhao, A. Jerebko, C. Florin, G.H. Valadez, L. Bogoni, and L. Moy. Supervised Learning from Multiple Experts: Whom to trust when everyone lies a bit. In *Proceedings of the 26th Annual International Conference on Machine Learning*. ACM New York, NY, USA, 2009.

[260] D. Rebholz-Schuhmann, M. Arregui, S. Gaudan, H. Kirsch, and A. Jimeno. Text processing through Web services: calling Whatizit. *Bioinformatics*, 24(2):296, 2008.

[261] D. Rebholz-Schuhmann, H. Kirsch, and F. Couto. Facts from text-is text mining ready to deliver? *PLoS Biol*, 3(2), 2005.

[262] D.J. Reiss, N.S. Baliga, and R. Bonneau. Integrated biclustering of heterogeneous genome-wide datasets for the inference of global regulatory networks. *BMC bioinformatics*, 7(1):280, 2006.

[263] F. Rinaldi, G. Schneider, K. Kaljurand, J. Dowdall, C. Andronis, A. Persidis, and O. Konstanti. Mining relations in the GENIA corpus. *Proc. 2nd European Workshop on Data Mining and Text Mining for Bioinformatics ECML/PKDD 2004*, pages 61–68, 2004.

[264] B. Rosario and M.A. Hearst. Multi-way relation classification: application to protein-protein interactions. In *Proceedings of the conference on Human Language Technology and Empirical Methods in Natural Language Processing*, page 739. Association for Computational Linguistics, 2005. http://biotext.berkeley.edu/data.html.

[265] B.C. Russell, A. Torralba, K.P. Murphy, and W.T. Freeman. LabelMe: a database and web-based tool for image annotation. *International Journal of Computer Vision*, 77(1):157–173, 2008.

[266] A. Rzhetsky, I. Iossifov, T. Koike, M. Krauthammer, P. Kra, M. Morris, H. Yu, P.A. Duboué, W. Weng, W.J. Wilbur, et al. GeneWays: a system for extracting, analyzing, visualizing, and integrating molecular pathway data. *Journal of Biomedical Informatics*, 37(1):43–53, 2004.

[267] D. Saad. *On-line learning in neural networks*. Cambridge Univ Pr, 1998. Chapter- Online Algorithms and Stochastic Approximations.

[268] Y. Saeys, S. Degroeve, and Y. Van de Peer. Feature Ranking Using an EDA-based Wrapper Approach. *Studies in Fuzziness and Soft Computing*, 192:243, 2006.

[269] Y. Saeys, I. Inza, and P. Larranaga. A review of feature selection techniques in bioinformatics. *Bioinformatics*, 23(19):2507, 2007.

[270] N. Salomonis, K. Hanspers, A.C. Zambon, K. Vranizan, S.C. Lawlor, K.D. Dahlquist, S.W. Doniger, J. Stuart, B.R. Conklin, and A.R. Pico. GenMAPP 2: new features and resources for pathway analysis. *BMC bioinformatics*, 8(1):217, 2007.

[271] F. Sanger, S.Ñicklen, and AR Coulson. DNA sequencing with chain-terminating inhibitors. *Proceedings of the National Academy of Sciences*, 74(12):5463, 1977.

[272] J. Saric, L.J. Jensen, R. Ouzounova, I. Rojas, and P. Bork. Extraction of regulatory gene/protein networks from Medline. *Bioinformatics*, 22(6):645, 2006.

[273] Y. Sasaki, S. Montemagni, P. Pezik, D. Rebholz-Schuhmann, J. McNaught, and S. Ananiadou. BioLexicon: A Lexical Resource for the Biology Domain. In *Proc. of the Third International Symposium on Semantic Mining in Biomedicine (SMBM 2008)*, 2008.

[274] Y. Sasaki, Y. Tsuruoka, J. McNaught, and S. Ananiadou. How to make the most of NE dictionaries in statistical NER. *BMC bioinformatics*, 9(Suppl 11):S5, 2008.

[275] E. Schadt, B. Zhang, and J. Zhu. Advances in systems biology are enhancing our understanding of disease and moving us closer to novel disease treatments. *Genetica*, 136(2):259–269, 2009.

[276] H. Schmid. Probabilistic part-of-speech tagging using decision trees. In *Proceedings of International Conference on New Methods in Language Processing*, volume 12. Manchester, UK, 1994.

[277] B. Scholkopf, A. Smola, and K.R. Muller. Nonlinear component analysis as a kernel eigenvalue problem. *Neural computation*, 10(5):1299–1319, 1998.

[278] SC Schuster. Next-generation sequencing transforms today's biology. *Nature methods*, 5(1):16, 2008.

[279] Burr Settles. Active learning literature survey. Computer Sciences Technical Report 1648, University of Wisconsin–Madison, 2009.

[280] N. Shah, C. Jonquet, A. Chiang, A. Butte, R. Chen, and M. Musen. Ontology-driven indexing of public datasets for translational bioinformatics. *BMC bioinformatics*, 10(Suppl 2):S1, 2009.

[281] R. Sharan, A. Maron-Katz, and R. Shamir. CLICK and EXPANDER: a system for clustering and visualizing gene expression data. *Bioinformatics*, 19(14):1787, 2003.

[282] H. Shatkay. Hairpins in bookstacks: information retrieval from biomedical text. *Briefings in bioinformatics*, 6(3):222, 2005.

[283] Q. Sheng, Y. Moreau, and B. De Moor. Biclustering microarray data by Gibbs sampling. *Bioinformatics-Oxford*, 19(2):196–205, 2003.

[284] J. Shi and J. Malik. Normalized cuts and image segmentation. *IEEE Transactions on pattern analysis and machine intelligence*, 22(8):888–905, 2000.

[285] D.D.K. Sleator and D. Temperley. Parsing English with a link grammar. *Arxiv preprint cmp-lg/9508004*, 1995. 3rd Int. Workshop on Parsing Technologies.

[286] Yanai I. Slonim DK. Getting started in gene expression microarray analysis. *PLoS Comput Biol*, 5(10):E1000543, 2009.

[287] LH Smith, L. Tanabe, T. Rindflesch, and WJ Wilbur. MedTag: a collection of biomedical annotations. In *Proceedings of the Workshop on Linking Biological Literature, Ontologies and Databases: Mining Biological Semantics*, page 32. Association for Computational Linguistics, 2005.

[288] G.K. Smyth and T. Speed. Normalization of cDNA microarray data. *Methods*, 31(4):265–273, 2003.

[289] R. Snow, B. O'Connor, D. Jurafsky, and A.Y. Ng. Cheap and fast—but is it good?: evaluating non-expert annotations for natural language tasks. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing*, pages 254–263. Association for Computational Linguistics, 2008.

[290] B. South, S. Shen, M. Jones, J. Garvin, M. Samore, W. Chapman, and A. Gundlapalli. Developing a manually annotated clinical document corpus to identify phenotypic information for inflammatory bowel disease. *BMC bioinformatics*, 10(Suppl 9):S12, 2009.

[291] I. Spasić, D. Schober, S.A. Sansone, D. Rebholz-Schuhmann, D. Kell, and N. Paton. Facilitating the development of controlled vocabularies for metabolomics technologies with text mining. *BMC bioinformatics*, 9(Suppl 5):S5, 2008.

[292] J. Sprenger, J.L. Fink, and R. Teasdale. Evaluation and comparison of mammalian subcellular localization prediction methods. *BMC bioinformatics*, 7(Suppl 5):S3, 2006.

[293] B.M. Sundheim and N. Chinchor. Named entity task definition, version 2.1. In *Proceedings of the Sixth Message Understanding Conference (MUC-6)*, pages 319–332, 1995.

[294] T. Suzuki, H. Yokoi, S. Fujita, and K. Takabayashi. Automatic DPC code selection from electronic medical records: text mining trial of discharge summary. *Methods of information in medicine*, 47(6):541–8, 2008.

[295] P. Tamayo, D. Slonim, J. Mesirov, Q. Zhu, S. Kitareewan, E. Dmitrovsky, E.S. Lander, and T.R. Golub. Interpreting patterns of gene expression with self-organizing maps: methods and application to hematopoietic differentiation. *Proceedings of the National Academy of Sciences of the United States of America*, 96(6):2907, 1999.

[296] F. Tan, X. Fu, Y. Zhang, and A.G. Bourgeois. A genetic algorithm-based method for feature subset selection. *Soft Computing-A Fusion of Foundations, Methodologies and Applications*, 12(2):111–120, 2008.

[297] L. Tanabe, N. Xie, L. Thom, W. Matten, and W.J. Wilbur. GENETAG: a tagged corpus for gene/protein named entity recognition. *BMC bioinformatics*, 6(Suppl 1):S3, 2005.

[298] A. Tanay, R. Sharan, and R. Shamir. Discovering statistically significant biclusters in gene expression data. *Bioinformatics*, 18:136–144, 2002.

[299] H. Tang, Y. Mukomel, and E. Fink. Diagnosis of ovarian cancer based on mass spectra of blood samples. In *Proceedings of the IEEE International Conference on Systems, Man and Cybernetics*, pages 3444–3450, 2004.

[300] L. Tari, C. Baral, and S. Kim. Fuzzy c-means clustering with prior biological knowledge. *Journal of Biomedical Informatics*, 42(1):74–81, 2009.

[301] J. Tarraga, I. Medina, J. Carbonell, J. Huerta-Cepas, P. Minguez, E. Alloza, F. Al-Shahrour, S. Vegas-Azcarate, S. Goetz, P. Escobar, et al. GEPAS, a web-based tool for microarray data analysis and interpretation. *Nucleic Acids Research*, 36(Web Server issue):W308, 2008.

[302] Y. Tateisi, A. Yakushiji, T. Ohta, and J. Tsujii. Syntax Annotation for the GENIA corpus. In *Second International Joint Conference on Natural Language Processing (IJCNLP'05)*, pages 222–227, 2005. http://www-tsujii.is.s.u-tokyo.ac.jp/GENIA/home/wiki.cgi.

[303] S. Tavazoie, J.D. Hughes, M.J. Campbell, R.J. Cho, and G.M. Church. Systematic determination of genetic network architecture. *Nature genetics*, 22:281–285, 1999.

[304] J.M. Temkin and M.R. Gilder. Extraction of protein interaction information from unstructured text using a context-free grammar. *Bioinformatics*, 19(16):2046, 2003.

[305] J. Thomas, D. Milward, C. Ouzounis, S. Pulman, and M. Carroll. Automatic extraction of protein interactions from scientific abstracts. In *Pacific Symposium on Biocomputing. Pacific Symposium on Biocomputing*, page 541, 2000.

[306] R. Tibshirani, G. Walther, and T. Hastie. Estimating the number of clusters in a data set via the gap statistic. *Journal of the Royal Statistical Society. Series B (Statistical Methodology)*, 63(2):411–423, 2001.

[307] P. Toronen. Selection of informative clusters from hierarchical cluster tree with gene classes. *BMC bioinformatics*, 5(1):32, 2004.

[308] O. Troyanskaya, M. Cantor, G. Sherlock, P. Brown, T. Hastie, R. Tibshirani, D. Botstein, and R.B. Altman. Missing value estimation methods for DNA microarrays. *Bioinformatics*, 17(6):520, 2001.

[309] G.C. Tseng. Penalized and weighted K-means for clustering with scattered objects and prior information in high-throughput biological data. *Bioinformatics*, 23(17):2247, 2007.

[310] G.C. Tseng, M.K. Oh, L. Rohlin, J.C. Liao, and W.H. Wong. Issues in cDNA microarray analysis: quality filtering, channel normalization, models of variations and assessment of gene effects. *Nucleic Acids Research*, 29(12):2549, 2001.

[311] Y. Tsuruoka and J. Tsujii. Improving the performance of dictionary-based approaches in protein name recognition. *Journal of Biomedical Informatics*, 37(6):461–470, 2004.

[312] J. Tuikkala, L. Elo, O.S. Nevalainen, and T. Aittokallio. Improving missing value estimation in microarray data with gene ontology. *Bioinformatics*, 22(5):566, 2006.

[313] H. Turner, T. Bailey, and W. Krzanowski. Improved biclustering of microarray data demonstrated through systematic performance tests. *Computational Statistics and Data Analysis*, 48(2):235–254, 2005.

[314] H.L. Turner, T.C. Bailey, W.J. Krzanowski, and C.A. Hemingway. Biclustering models for structured microarray data. *IEEE/ACM Transactions on Computational Biology and Bioinformatics*, pages 316–329, 2005.

[315] V.G. Tusher, R. Tibshirani, and G. Chu. Significance analysis of microarrays, apr 22 2008. US Patent 7,363,165.

[316] O. Uzuner. Recognizing Obesity and Comorbidities in Sparse Data. *Journal of the American Medical Informatics Association*, 16(4):561–570, 2009.

[317] O. Uzuner, Y. Luo, and P. Szolovits. Evaluating the state-of-the-art in automatic de-identification. *Journal of the American Medical Informatics Association*, 14(5):550–563, 2007.

[318] O. Uzuner, T.C. Sibanda, Y. Luo, and P. Szolovits. A de-identifier for medical discharge summaries. *Artificial intelligence in medicine*, 2007.

[319] Kohane I Uzuner O, Szolovits P. i2b2 workshop on natural language processing challenges for clinical records. . In *Proceedings of the Fall Symposium of the AMIA.*, 2006.

[320] Kohane I (organizers) Uzuner O, Szolovits P. 2nd i2b2 Shared-Task and Workshop Challenges in Natural Language Processing for Clinical Data. Obesity Challenge (A Shared-Task on Obesity): Who's obese and what co-morbidities do they (definitely/likely) have?., 2008. https://www.i2b2.org/NLP/.

[321] V. van Noort, B. Snel, and M.A. Huynen. Predicting gene function by conserved co-expression. *Trends in Genetics*, 19(5):238–242, 2003.

[322] C. Von Mering, L.J. Jensen, M. Kuhn, S. Chaffron, T. Doerks, B. Kruger, B. Snel, and P. Bork. STRING 7–recent developments in the integration and prediction of protein interactions. *Nucleic acids research*, 35(Database issue):D358, 2007.

[323] H. Wang, W. Wang, J. Yang, and P.S. Yu. Clustering by pattern similarity in large data sets. In *Proceedings of the 2002 ACM SIGMOD international conference on Management of data*, page 405. ACM, 2002.

[324] J.Y.W.W.H. Wang and P. Yu. δ-clusters: capturing subspace correlation in a large dataset. In *Data Engineering, 2002. Proceedings. 18th International Conference on*, pages 517–528, 2002.

[325] X. Wang, A. Li, Z. Jiang, and H. Feng. Missing value estimation for DNA microarray gene expression data by Support Vector Regression imputation and orthogonal coding scheme. *BMC bioinformatics*, 7(1):32, 2006.

[326] Y. Wang, J.D. Kim, R. Saetre, S. Pyysalo, and J. Tsujii. Investigating heterogeneous protein annotations toward cross-corpora utilization. *BMC bioinformatics*, 10(1):403, 2009.

[327] J.D. Watson and F.H.C. Crick. A structure for deoxyribose nucleic acid. *Nature*, 171:737–738, 1953.

[328] B. Wellner, M. Huyck, S. Mardis, J. Aberdeen, A. Morgan, L. Peshkin, A. Yeh, J. Hitzeman, and L. Hirschman. Rapidly retargetable approaches to de-identification in medical records. *Journal of the American Medical Informatics Association*, 14(5):564–573, 2007.

[329] M. West, C. Blanchette, H. Dressman, E. Huang, S. Ishida, R. Spang, H. Zuzan, J.A. Olson, J.R. Marks, and J.R. Nevins. Predicting the clinical status of human breast cancer by using gene expression profiles. *Proceedings of the National Academy of Sciences of the United States of America*, 98(20):11462, 2001.

[330] J. Weston, C. Leslie, E. Ie, D. Zhou, A. Elisseeff, and W.S. Noble. Semi-supervised protein classification using cluster kernels. *Bioinformatics*, 21(15):3241, 2005.

[331] J. Wilbur, L. Smith, and L. Tanabe. Biocreative 2. gene mention task. In *Proceedings of the second biocreative challenge evaluation workshop*, pages 7–16, 2007.

[332] R. Xu et al. Survey of clustering algorithms. *IEEE Transactions on neural networks*, 16(3):645–678, 2005.

[333] et al. Y. Kluger, R. Basri. Spectral biclustering of microarray cancer data: Co-clustering genes and conditions. *Genome Research*, 13:703–716, 2003.

[334] J. Yang, H. Wang, W. Wang, and P. Yu. Enhanced Biclustering on Expression Data. In *Proceedings of the 3rd IEEE Symposium on BioInformatics and BioEngineering*, page 321. IEEE Computer Society, 2003.

[335] Z. Yang, H. Lin, and Y. Li. BioPPISVMExtractor: A Protein-Protein Interaction Extractor for Biomedical Literature Using SVM and Rich Feature Sets. *Journal of biomedical informatics*, 2009.

[336] KY Yeung, C. Fraley, A. Murua, AE Raftery, and WL Ruzzo. Model-based clustering and data transformations for gene expression data. *Bioinformatics*, 17(10):977, 2001.

[337] KY Yeung, DR Haynor, and WL Ruzzo. Validating clustering for gene expression data. *Bioinformatics*, 17(4):309, 2001.

[338] G. Yona, W. Dirks, and S. Rahman. Comparing algorithms for clustering of expression data: how to assess gene clusters. *Methods in molecular biology (Clifton, NJ)*, 541:479, 2009.

[339] C.S. Yu, Y.C. Chen, C.H. Lu, and J.K. Hwang. Prediction of protein subcellular localization. *Proteins: Structure, Function and Bioinformatics*, 64(3):643–651, 2006.

[340] LA Zadeh. Fuzzy sets*. *Information and control*, 8(3):338–353, 1965.

[341] B.R. Zeeberg, W. Feng, G. Wang, M.D. Wang, A.T. Fojo, M. Sunshine, S.Ñarasimhan, D.W. Kane, W.C. Reinhold, S. Lababidi, et al. GoMiner: a resource for biological interpretation of genomic and proteomic data. *Genome Biol*, 4(4):R28, 2003.

[342] Q.T. Zeng, S. Goryachev, S. Weiss, M. Sordo, S.N. Murphy, and R. Lazarus. Extracting principal diagnosis, co-morbidity and smoking status for asthma research: evaluation of a natural language processing system. *BMC medical informatics and decision making*, 6(1):30, 2006.

[343] J.S. Zhang and Y.W. Leung. Improved possibilistic C-means clustering algorithms. *IEEE Transactions on Fuzzy Systems*, 12(2):209–217, 2004.

[344] Q. Zhang and Y. Zhang. Hierarchical clustering of gene expression profiles with graphics hardware acceleration. *Pattern Recognition Letters*, 27(6):676–681, 2006.

[345] Z. Zhang, K.H. Cheung, and J.P. Townsend. Bringing Web 2.0 to bioinformatics. *Briefings in Bioinformatics*, 10(1):1–10, 2009.

[346] L. Zhao and MJ Zaki. MicroCluster: efficient deterministic biclustering of microarray data. *IEEE Intelligent Systems*, 20(6):40–49, 2005.

[347] D. Zhu. Semi-supervised gene shaving method for predicting low variation biological pathways from genome-wide data. *BMC bioinformatics*, 10(Suppl 1):S54, 2009.

[348] X. Zhu. Semi-supervised learning literature survey. Technical report, Computer Science, University of Wisconsin-Madison, 2007. http://pages.cs.wisc.edu/ jerryzhu/pub/ssl_survey.pdf.

[349] E. Zitzler, M. Laumanns, L. Thiele, et al. SPEA2: Improving the strength Pareto evolutionary algorithm. In *EUROGEN*, pages 95–100, 2001.

[350] R. Zukiel, S.Ñowak, A.M. Barciszewska, I. Gawronska, G. Keith, and M.Z. Barciszewska. A simple epigenetic method for the diagnosis and classification of brain tumors. *Molecular Cancer Research*, 2(3):196, 2004.

[351] P. Zweigenbaum, D. Demner-Fushman, H. Yu, and K.B. Cohen. Frontiers of biomedical text mining: current progress. *Briefings in Bioinformatics*, 8(5):358, 2007.