



UNIVERSIDAD DE MÁLAGA

Escuela Técnica Superior de Ingeniería Informática

Departamento de Lenguajes y Ciencias de la Computación

Programa de Doctorado en Tecnologías Informáticas

TESIS DOCTORAL

De la Información al Conocimiento

**Aplicaciones basadas en implicaciones y
computación paralela**

Autor: D. Fernando Benito Picazo

Directores: Dr. D. Manuel Nicolás Enciso García-Oliveros


Dr. D. Carlos Manuel Rossi Jiménez

Málaga, 2018



UNIVERSIDAD
DE MÁLAGA

AUTOR: Fernando Benito Picazo

 <http://orcid.org/0000-0001-9679-1043>

EDITA: Publicaciones y Divulgación Científica. Universidad de Málaga



Esta obra está bajo una licencia de Creative Commons Reconocimiento-NoComercial-SinObraDerivada 4.0 Internacional:

<http://creativecommons.org/licenses/by-nc-nd/4.0/legalcode>

Cualquier parte de esta obra se puede reproducir sin autorización pero con el reconocimiento y atribución de los autores.

No se puede hacer uso comercial de la obra y no se puede alterar, transformar o hacer obras derivadas.

Esta Tesis Doctoral está depositada en el Repositorio Institucional de la Universidad de Málaga (RIUMA): riuma.uma.es





UNIVERSIDAD
DE MÁLAGA



UNIVERSIDAD
DE MÁLAGA

UNIVERSIDAD DE MÁLAGA

Escuela Técnica Superior de Ingeniería Informática

Departamento de Lenguajes y Ciencias de la Computación

Dr. D. Manuel Nicolás Enciso García-Oliveros, Catedrático de Escuela Universitaria del Departamento de Lenguajes y Ciencias de la Computación de la Universidad de Málaga,

Dr. D. Carlos Manuel Rossi Jiménez, Catedrático de Escuela Universitaria del Departamento de Lenguajes y Ciencias de la Computación de la Universidad de Málaga

HACEN CONSTAR QUE:

D. Fernando Benito Picazo, Ingeniero en Informática, ha realizado en el Departamento de Lenguajes y Ciencias de la Computación de la Universidad de Málaga, bajo nuestra dirección, el trabajo de investigación correspondiente a su Tesis Doctoral titulado:

De la Información al Conocimiento

Aplicaciones basadas en implicaciones y computación paralela

Revisado el presente trabajo, estimamos que puede ser presentado al tribunal que ha de juzgarlo, y autorizamos la presentación de esta Tesis Doctoral en la Universidad de Málaga.

Málaga, Diciembre de 2018

Dr. Manuel N. Enciso García-Oliveros Dr. Carlos M. Rossi Jiménez



UNIVERSIDAD
DE MÁLAGA

*A mi querida Aurora,
a mi padre y a mi madre,
a mi hermano,
por el apoyo y la confianza que
siempre me habéis demostrado.*

¡Y a la alegría de la casa, la Elsita!



UNIVERSIDAD
DE MÁLAGA

*En todo objetivo conseguido hay siempre una especial tristeza,
en el conocimiento de que una meta largamente deseada
se ha logrado al fin, y que la vida tiene entonces que ser
moldeada y encaminada en busca de nuevos fines.*

La Ciudad y las Estrellas

A. C. Clarke



UNIVERSIDAD
DE MÁLAGA

Agradecimientos

A menudo he escuchado que esta es la parte que más difícil resulta redactar; coincido con ello, pero también tengo claro que es la que más he disfrutado escribiendo.

Ahora que ya se termina esta etapa de mi vida con la consecución de esta tesis doctoral, es momento de hacer balance y de agradecer a las personas que han intervenido en este gran logro; porque es también gracias a ellos.

En primer lugar, quiero agradecer a mis queridos directores, Dr. Manuel Enciso y Dr. Carlos Rossi, la dirección de la tesis y todo lo que me habéis enseñado, pero en especial, vuestro trato, con el que siempre me habéis hecho sentir uno de vosotros, y donde la acogida siempre ha sido la mejor; por todo ello, gracias.

Manolo, me has dirigido: el proyecto fin de carrera de la ingeniería técnica de sistemas, el proyecto fin de carrera de la ingeniería superior, el trabajo fin de máster, y ahora, la tesis doctoral. Quiero decirte que me siento orgulloso de ello, y que si tuviera que volver a recorrer todo ese camino, volvería a pedirte que estuvieras de nuevo a mi lado; por todo ello, gracias.

Carlos, lo primero es decirte que ojalá nos hubiéramos conocido antes. Además de por la dirección de la tesis, también quiero darte las gracias por haber buscado siempre posibilidad de financiación para que pudiera afrontar este periodo de investigación con un respaldo; por todo ello, gracias.

Gracias a mis coautores: Dr. Pablo Cordero, Dr. Ángel Mora, Dr. An-

tonio Guevara, ha sido un placer publicar a vuestro lado; a la Dra. Llanos Mora como tutora de tesis; al Dr. Antonio Vallecillo por buscar un contacto en el momento preciso; al Dr. Darío Guerrero y al Dr. Rafael Larrosa por el apoyo en el Centro de Supercomputación.

Quiero hacer mención especial al Dr. Sergio Gálvez. Sergio, ha sido una gran suerte tenerte cerca para dudas y consejos, y en realidad, por el simple hecho de conversar contigo. Siempre me has apoyado y valorado mucho, y eso significa mucho para mí; por todo ello, gracias.

Ahora, quiero darte las gracias a ti, mi querida Aurora, por ser mi apoyo y ánimo diario e incondicional, por quererme siempre cada día y entender mi insaciable inquietud, y por haber vivido juntos todos estos éxitos que la vida nos está deparando, que seguro, a tu lado, serán muchos más.

Lo primero que quiero decirle a mi padre y a mi madre es: lo he conseguido. Gracias por darme el mejor entorno familiar posible, por vuestro interminable esfuerzo y por inculcarme la vital importancia de la educación y el saber. Ahora es momento de recoger los frutos y disfrutar de lo que hemos conseguido todos juntos.

Gracias también a ti, hermano, por ser la persona que más me ha enseñado en la vida y de quien sigo aprendiendo a cada momento juntos. Gracias por estar siempre disponible y por encauzarme y guiarme por el infinito camino de la Ciencia.

Quiero hacer un guiño a mis profesores de la niñez, D. Justo, D. Julián, D. Luis y D. Andrés, por construir a tan temprana edad, unos sólidos cimientos sobre los que crecer.

Seguramente olvide alguien, no obstante, gracias a toda persona que hasta el día de hoy me haya enseñado cualquier cosa, pues para mí, lo más importante es el *Conocimiento*.

Índice general

Publicaciones	v
1. Introducción	1
1.1. Claves Minimales	6
1.2. Generadores Minimales	12
1.3. Sistemas de Recomendación Conversacionales	15
1.4. Verificación de los Resultados	20
1.5. Estructura de la Tesis	23
2. Preliminares	27
2.1. Análisis Formal de Conceptos	30
2.1.1. Contextos Formales	30
2.1.2. Operadores de Derivación	32
2.1.3. Conceptos Formales	34
2.1.4. Retículo de Conceptos	35
2.1.5. Sistema de Implicaciones	37
2.2. Bases de Datos Relacionales	39
2.2.1. Dependencias Funcionales	41
2.3. Lógica de Implicaciones	43
2.3.1. Axiomas de Armstrong	43
2.3.2. Lógica de Simplificación	44
2.4. Razonamiento Automático	46
2.4.1. Algoritmos para el Cálculo del Cierre	47



3. Claves Minimales	53
4. Generadores Minimales	59
5. Sistemas de Recomendación Conversacionales	65
6. Conclusiones y Trabajos Futuros	71
6.1. Conclusiones	74
6.2. Trabajos Futuros	80
Índice Alfabético	83
Índice de Figuras	85
Índice de Tablas	87
Anexo	89
A. Closed sets enumeration: a logical approach	91
B. Conversational recommendation to avoid the cold-start problem	95
C. Keys for the fusion of heterogeneous information	99
D. Increasing the Efficiency of Minimal Key Enumeration Methods by Means of Parallelism	103
Bibliografía	107

Publicaciones

A continuación se expone una lista de los trabajos que han sido publicados como resultado de la investigación llevada a cabo a lo largo de esta tesis doctoral. Estas publicaciones avalan el trabajo realizado poniendo de manifiesto tanto su interés como su validez científica.

Revistas

- (I) Fernando Benito-Picazo, Pablo Cordero, Manuel Enciso, Ángel Mora. *Minimal generators, an affordable approach by means of massive computation*. The Journal of Supercomputing, Springer, 2018.

DOI: 10.1007/s11227-018-2453-z

Factor de impacto en J.C.R. 2017: 1,532. Posición 43 de 103 (Q2) en la categoría: 'Computer Science, Theory & Methods'.

- (II) Fernando Benito-Picazo, Manuel Enciso, Carlos Rossi, Antonio Guevara. *Enhancing the conversational process by using a logical closure operator in phenotypes implications*. Mathematical Methods in the Applied Sciences, John Wiley & Sons Ltd, 2017.

DOI: 10.1002/mma.4338

Factor de impacto en J.C.R. 2017: 1,18. Posición 91 de 252 (Q2) en la categoría: 'Mathematics, Applied'.

- (III) Fernando Benito-Picazo, Pablo Cordero, Manuel Enciso, Ángel Mora. *Reducing the search space by closure and simplification paradigms*. A

parallel key finding method. The Journal of Supercomputing, Springer, 2016.

DOI: 10.1007/s11227-016-1622-1

Factor de impacto en J.C.R. 2016: 1,349. Posición 52 de 104 (Q2) en la categoría: ‘Computer Science, Theory & Methods’.

Congresos Internacionales

- Fernando Benito-Picazo, Pablo Cordero, Manuel Enciso, Ángel Mora. *Closed sets enumeration: a logical approach*. Proceedings of the Seventeenth International Conference on Computational and Mathematical Methods in Science and Engineering, CMMSE, 2017. Cádiz, Spain, July 4-8, pp. 287-292, ISBN: 978-84-617-8694-7.
- Fernando Benito-Picazo, Manuel Enciso, Carlos Rossi, Antonio Guevara. *Conversational recommendation to avoid the cold-start problem*. Proceedings of the Sixteenth International Conference on Computational and Mathematical Methods in Science and Engineering, CMMSE, 2016. Cádiz, Spain, July 4-8, pp. 184-190, ISBN: 978-84-608-6082-2.
- Fernando Benito-Picazo, Pablo Cordero, Manuel Enciso, Ángel Mora. *Keys for the fusion of heterogeneous information*. Proceedings of the Fifteenth International Conference on Computational and Mathematical Methods in Science and Engineering, CMMSE, 2015. Cádiz, Spain, July 6-10, pp. 201-211, ISBN: 978-84-617-2230-3.
- Fernando Benito-Picazo, Pablo Cordero, Manuel Enciso, Ángel Mora. *Increasing the Efficiency of Minimal Key Enumeration Methods by Means of Parallelism*. Proceedings of the 9th International Conference on Software Engineering and Applications, ICSoft-EA, Vienna, Austria, August 29-31, 2014, pp. 512-517.

DOI: 10.5220/0005108205120517

Workshops

- Fernando Benito-Picazo. *Parallelism in the search of minimal keys from implications using tableaux methods*. Workshop: Lógica, Lenguaje e Información. Dpto. Matemática Aplicada. Unidad de Investigación en Lógica, Lenguaje e Información, Andalucía Tech. Universidad de Málaga, Noviembre 2014.

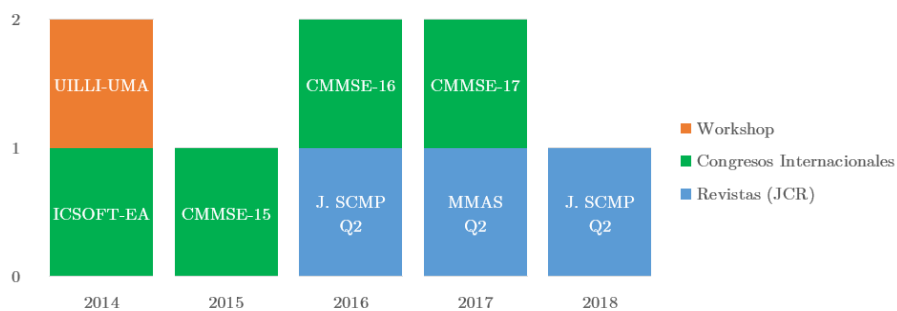


Figura 1: Producción científica



UNIVERSIDAD
DE MÁLAGA

Capítulo 1

Introducción





UNIVERSIDAD
DE MÁLAGA

—Empieza por el principio —dijo el Rey con gravedad—
y sigue hasta llegar al final; allí, te paras.

Alicia en el país de las maravillas

L. Carroll

La gestión de la información es uno de los pilares esenciales de la Ingeniería Informática. No es de extrañar, por tanto, que conforme un amplio campo de investigación y conocimiento donde diversas disciplinas como las Matemáticas, la Lógica y la Computación actúen conjuntamente para alcanzar mejores sinergias.

Dentro de este ámbito y con la intención de hacer aportaciones en campos de la Ingeniería Informática como son las bases de datos y los sistemas de recomendación, esta tesis doctoral toma como principal base teórica el Análisis Formal de Conceptos (FCA, por sus siglas en inglés: *Formal Concept Analysis*), y más concretamente, una de sus herramientas fundamentales: los conjuntos de implicaciones. La gestión inteligente de estos elementos mediante técnicas lógicas y computacionales confieren una alternativa para superar obstáculos en los campos mencionados.

FCA es una teoría matemática y una metodología que permite derivar una jerarquía de conceptos a partir de una colección de objetos, sus atribu-

tos y las relaciones entre ellos. De esta forma, el propósito es poder representar y organizar la información de manera más cercana al pensamiento humano sin perder rigor científico. En este sentido se enmarca la cita de Rudolf Wille: “*El objetivo y el significado del FCA como teoría matemática sobre conceptos y sus jerarquías es apoyar la comunicación racional entre seres humanos mediante el desarrollo matemático de estructuras conceptuales apropiadas que se puedan manipular con la lógica.*” [127].

El término FCA fue acuñado por Wille en 1984 culminando años más tarde con la publicación más citada al respecto en colaboración con Bernhard Ganter [46]. Desde entonces, FCA se ha aplicado con éxito en diferentes disciplinas, como por ejemplo: biología celular [39], genética [119], ingeniería del software [65, 95], medicina [104], derecho [82], etc.

FCA parte de una representación de conjuntos de objetos y atributos por medio de tablas de datos. Estas tablas se denominan contextos formales y representan las relaciones binarias entre esos objetos y atributos. A partir de ahí, se generan dos herramientas básicas para representar el conocimiento: los retículos de conceptos y los conjuntos de implicaciones. Dichas herramientas además, son representaciones equivalentes del conocimiento descrito en el contexto formal.

Desde hace años, existen en la literatura estudios [67, 91] donde se han investigado y comparado diferentes algoritmos para obtener el retículo de conceptos a partir de un conjunto de datos (en adelante *dataset* por su nomenclatura habitual en el campo). Muchos de ellos toman como base uno de los algoritmos más conocidos a tal efecto, el denominado por Wille y Ganter como *NextClosure* [46].

Por otro lado está el conjunto de implicaciones. Las implicaciones pueden considerarse *grosso modo* como reglas del tipo *si-entonces*, que representan un concepto muy intuitivo: cuando se verifica una premisa, entonces se cumple una conclusión. Esta idea básica se utiliza con diferentes interpretaciones en numerosos campos de conocimiento. Así, en la teoría relacional se interpretan como dependencias funcionales (DFs) [22], en FCA como implicaciones [46], etcétera.

No obstante, también existen ciertas desventajas a la hora de trabajar

con retículos e implicaciones, de hecho, la propia extracción del conjunto completo de implicaciones de un *dataset* es una tarea que presenta una complejidad exponencial, sin embargo, es conveniente destacar que no es competencia de este trabajo el estudio de técnicas de extracción de implicaciones (lo cual es más una tarea de minería de datos), sino que la intención es partir del conjunto de implicaciones para trabajar con él. A este respecto, se pueden consultar trabajos ampliamente citados en la literatura en relación a la extracción de implicaciones a partir de *datasets* [57, 131].

Trabajar con conjuntos de implicaciones permite utilizar técnicas de razonamiento automático basadas en la lógica. Este hecho fundamenta el objetivo de esta tesis doctoral, que principalmente consiste en, utilizando los conjuntos de implicaciones, aplicar mecanismos lógicos para realizar un tratamiento eficiente de la información.

Como se verá en el Capítulo 2, la aproximación a través de la lógica es posible gracias a sistemas axiomáticos correctos y completos como los axiomas de Armstrong [4] y la Lógica de Simplificación [26] (SL, por sus siglas en inglés: *Simplification Logic*). Estos métodos aplicados sobre conjuntos de implicaciones se utilizan en esta tesis doctoral sobre tres áreas de investigación: claves minimales, generadores minimales y sistemas de recomendación conversacionales.

Se anticipa que, en cada uno de los casos, se va a aprovechar la información subyacente al conjunto de implicaciones para realizar novedosas aproximaciones que permitan abordar problemas presentes en esos ámbitos. Los resultados obtenidos se sustentan por una amplia gama de experimentos, en los cuales se ha utilizado tanto información real como sintética (información generada de forma aleatoria) y donde la computación paralela llevada a cabo en entornos de supercomputación ha desempeñado un papel crucial. Como se verá más adelante, para el primer caso, el trabajo se centra en el uso de DFs mientras que para el segundo y tercero el núcleo son implicaciones.

Dicho esto, se retoma el texto pasando a introducir los tres campos de aplicación donde se han utilizado los conjuntos de implicaciones.

1.1. Claves Minimales

El concepto de clave es fundamental en cualquier modelo de datos, incluyendo el modelo de datos relacional de Codd [24]. Una clave de un esquema relacional está compuesta por un subconjunto de atributos que identifican a cada uno de los elementos de una relación. Representan el *dominio* de una determinada función cuya *imagen* es la totalidad del conjunto de atributos. Así, en un esquema de bases de datos relacional, una clave permite identificar cada fila de una tabla, impidiendo que exista más de una fila con la misma información y puede representarse por medio de una DF [24] hacia todo el conjunto de atributos. Debido a ello en los sistemas gestores de bases de datos, las restricciones de clave son implementadas usando restricciones de unicidad (*unique*) sobre el subconjunto de atributos que forman la clave.

Las DFs especifican una relación entre dos subconjuntos de atributos, e.g. A y B , representada como $A \rightarrow B$, que asegura que para cualesquiera dos tuplas de una tabla de datos, si los valores de sus atributos de A coinciden, entonces también han de coincidir los de B . Si bien la noción de DF se verá con mayor detalle en la Sección 2.2, se adelanta el siguiente ejemplo básico 1.1.1 tanto para mostrar ejemplos de DFs como para ilustrar el concepto de clave.

Ejemplo 1.1.1. *Supongamos que disponemos de la siguiente tabla con información que relaciona títulos de películas, actores, países, directores, nacionalidad, valoración y años de estreno:*

<i>Título</i>	<i>Año</i>	<i>País</i>	<i>Director</i>	<i>Nacionalidad</i>	<i>Actor</i>	<i>Valoración</i>
<i>Pulp Fiction</i>	1994	USA	<i>Tarantino</i>	USA	<i>J. Travolta</i>	8
<i>Pulp Fiction</i>	1994	USA	<i>Tarantino</i>	USA	<i>U. Thurman</i>	9
<i>Pulp Fiction</i>	1994	USA	<i>Tarantino</i>	USA	<i>S. Jackson</i>	8
<i>King Kong</i>	2005	NZ	<i>Jackson</i>	NZ	<i>N. Watts</i>	9
<i>King Kong</i>	2005	NZ	<i>Jackson</i>	NZ	<i>J. Black</i>	6
<i>King Kong</i>	1976	USA	<i>Laurentiis</i>	IT	<i>J. Lange</i>	7
<i>King Kong</i>	1976	USA	<i>Laurentiis</i>	IT	<i>J. Bridges</i>	6
<i>Django Unchained</i>	2012	USA	<i>Tarantino</i>	USA	<i>J. Foxx</i>	8
<i>Django Unchained</i>	2012	USA	<i>Tarantino</i>	USA	<i>S. Jackson</i>	9
<i>Blade Runner</i>	1982	USA	<i>Scott</i>	UK	<i>H. Ford</i>	9
<i>Blade Runner</i>	2017	USA	<i>Villeneuve</i>	CAN	<i>H. Ford</i>	6

Entonces, un ejemplo sencillo de DF que se puede extraer de esta información es:

Director → *Nacionalidad*

Esta tabla tiene una única clave: { Título, Año, Actor} que corresponde con el conjunto de atributos necesario para identificar cualquier tupla de la relación.

La identificación de las claves de una determinada relación es una tarea crucial para muchas áreas de tratamiento de la información: modelos de datos [112], optimización de consultas [63], indexado [78], enlazado de datos [90], etc. Como muestras de esta importancia, es posible encontrar numerosas citas en la literatura, entre las que se pueden destacar las siguientes. En [113], los autores afirman que: “*la identificación de claves es una tarea fundamental en muchas áreas de la gestión moderna de datos, incluyendo modelado de datos, optimización de consultas (proporciona un optimizador de consultas con nuevas rutas de acceso que pueden conducir a mejoras sustanciales en el procesado de consultas), indexación (permite al administrador de la base de datos mejorar la eficiencia del acceso a los datos a través de técnicas como la partición de datos o la creación de índices y vistas), detección de anomalías e integración de datos*”. En [94] los autores delimitan el problema manifestando: “*establecer enlaces semánticos entre los elementos de datos puede ser realmente útil, ya que permite a los rastreadores, navegadores y aplicaciones combinar información de diferentes fuentes.*”.

Como refleja el contenido de esta sección, es evidente la importancia manifiesta de averiguar las claves de una relación, sin embargo, esta labor no está exenta de dificultades, por ello, el trabajo realizado en esta parte de la tesis ha consistido en proponer, diseñar e implementar métodos para afrontar el problema de la búsqueda de claves, el cual se presenta a continuación.

El Problema de la Búsqueda de Claves

El problema de la búsqueda de claves consiste en encontrar todos los subconjuntos de atributos que componen una clave minimal¹ a partir de un conjunto de DFs. Es un campo de estudio con décadas de antigüedad como puede observarse en [106], o en [40], donde las claves se estudiaron dentro del ámbito de la matriz de implicaciones.

El cálculo de todas las claves minimales representa un problema complejo. En [74,133] se incluyen resultados interesantes acerca de la complejidad del problema; los autores demuestran que el número de claves está limitado por el factorial del número de dependencias, por tanto, no existe un algoritmo que resuelva el problema en tiempo polinómico. En definitiva, es un problema NP-completo decidir si existe una clave de tamaño a lo sumo k dado un conjunto de DFs.

Por otro lado, en [27], los autores muestran cómo el problema de las claves minimales en las bases de datos tiene su análogo en FCA, donde el papel de las DFs se trata como implicaciones de atributos. En ese artículo, el problema de las claves minimales se presentó desde un punto de vista lógico y para ello, se empleó un sistema axiomático, que los autores denominaron SL_{FD} (por sus siglas en inglés: *Simplification Logic for Functional Dependencies*) [26], para gestionar las DFs y las implicaciones.

Las principales referencias sobre el problema de la búsqueda de claves apuntan al trabajo de Lucchesi y Osborn en [74] donde presentan un algoritmo para calcular todas las claves. Por otro lado, Saiedian y Spencer [105] presentaron un algoritmo usando grafos con atributos para encontrar todas las claves posibles de un esquema de base de datos relacional. No obstante, demostraron que sólo podía aplicarse cuando el grafo de DFs no estuviera fuertemente conectado. Es reseñable también el trabajo de Zhang [136] en el cual se utilizan mapas de Karnaugh [62] para calcular todas las claves. Existen más trabajos sobre el problema del cálculo de las claves minimales

¹Se acuña el término *minimal* para referirnos a una clave en la que todos y cada uno de los atributos que la forman son imprescindibles para mantener su naturaleza de clave, es decir, no contiene ningún atributo superfluo.

como son [113,128].

Algoritmos para el Cálculo de Claves

El objetivo de esta parte de la tesis se centra en los algoritmos de búsqueda de claves basados en la lógica, y más específicamente, en aquellos que utilizan el paradigma de tableaux [88,103] como sistema de inferencia.

De forma muy general, se puede decir que los métodos tipo tableaux representan el espacio de búsqueda como un árbol, donde sus hojas contienen las soluciones (claves). El proceso de construcción del árbol comienza con una raíz inicial y desde allí, mediante la utilización de unas reglas de inferencia, se generan nuevas ramas del árbol etiquetadas con nodos que representan instancias más simples del nodo padre. Debido a esta característica, las comparaciones entre estos métodos se pueden realizar fácilmente ya que su eficiencia va de la mano del tamaño del árbol de búsqueda generado. La mayor ventaja de este proceso es su versatilidad, ya que el desarrollo de nuevos métodos se reduce en gran parte a cambiar las reglas de inferencia.

Esto conduce a un punto de partida fundamental, los estudios de R. Wastl (Universidad de Wurzburg, Alemania) [124,125] donde se introduce por primera vez un sistema de inferencia de tipo Hilbert para averiguar todas las claves de un esquema relacional. Básicamente, se parte de una raíz para cuyo cálculo se ha aplicado una regla de inferencia $\mathbb{K}1$ y a partir de ahí, se van construyendo las diferentes ramas del árbol mediante la aplicación de una segunda regla de inferencia $\mathbb{K}2$ al conjunto de DFs (véase [124,125] para más detalles).

Siguiendo esta línea, en [25] los autores abordan el problema de la búsqueda de claves utilizando un sistema de inferencia basado en la lógica SL_{FD} [26], demostrando como el árbol del espacio de búsqueda que se genera lleva a sobrepasar las capacidades computacionales de ordenadores corrientes hoy día, incluso para problemas pequeños. En [26] los autores muestran la equivalencia entre SL_{FD} y los axiomas de Armstrong [4] junto con un algoritmo para calcular el cierre de un conjunto de atributos.

Más tarde, en [27], los autores introdujeron el método SST (por sus siglas en inglés, *Strong Simplification Tableaux*) para calcular todas las claves minimales usando una estrategia de estilo tableaux, abriendo la puerta a incorporar el paralelismo en su implementación.

El método SST está basado en la lógica de simplificación y sus equivalencias, añadiendo además, el test de minimalidad para aumentar la eficiencia. De esta forma, SST evita la apertura de ramas adicionales del árbol, por lo que el espacio de búsqueda se vuelve más reducido, logrando un gran rendimiento en comparación con sus predecesores como puede comprobarse en el amplio estudio realizado sobre el método en [10].

Por otro lado, el nuevo operador de cierre definido en [87] tiene una característica fundamental que lo convierte en una novedosa alternativa frente a los métodos clásicos [75] y es la siguiente. Además del conjunto de atributos que se deriva de la aplicación del operador de cierre al conjunto de implicaciones, el método proporciona un subconjunto Σ' de implicaciones del conjunto Σ original que engloba la información que ha quedado fuera del cierre.

Tomando como base esos trabajos anteriores y con el apoyo del sistema axiomático de la lógica SL_{FD} (véase Sección 2.3.2), en esta tesis se presenta un nuevo método llamado *Closure Keys (CK)*. Este nuevo método incorpora un mecanismo eficiente de poda que utiliza el método de cierre basado en SL_{FD} (ver Sección 2.4.1) para mejorar el rendimiento del método SST.

Una propiedad muy interesante de los métodos basados en tableaux (como lo son los métodos SST y CK) es la generación de subproblemas independientes los unos de los otros a partir del problema original. De esta forma, se alcanza otro objetivo fundamental de esta tesis, que consiste en utilizar las técnicas lógicas sobre una implementación paralela de los métodos de búsqueda de claves que, mediante el uso de recursos de supercomputación, permitan alcanzar resultados en un tiempo razonable.

Por nuestra parte, en [11] ya se presentó una primera aproximación a la paralelización del método de Wastl [124, 125] y el algoritmo de claves [25], donde se muestra cómo el paralelismo puede integrarse de forma natural en los métodos basados en tableaux. Siguiendo la línea de estos trabajos,

en esta tesis se ha llevado a cabo el estudio y diseño de los métodos SST y CK, y posteriormente, se han desarrollado también las implementaciones de los algoritmos en sus versiones secuenciales y paralelas, basándose estas últimas en el paradigma *MapReduce* [35].

Para la labor de computación de alto rendimiento, se ha trabajado intensamente con el Centro de Supercomputación y Bioinnovación de la Universidad de Málaga². La posibilidad de tratar con este centro ha proporcionado dos beneficios fundamentales: por un lado, se ha alcanzado una elevada pericia para trabajar en entornos de computación de alto rendimiento (HPC, por sus siglas en inglés: *High Performance Computing*) y para realizar implementaciones que aprovechen una alta cantidad de recursos, y por otro lado, ha permitido obtener resultados empíricos sobre experimentos utilizando estrategias paralelas que han desembocado en contribuciones científicas [11, 13] y que habría sido imposible conseguir en la actualidad sin contar con tales recursos computacionales.

Básicamente, el algoritmo paralelo de búsqueda de claves se divide en dos partes principales. Utiliza una primera fase en la que se realiza una expansión del árbol de búsqueda trabajando sobre el problema original y aplicando sucesivamente las reglas de inferencia y el algoritmo del cierre lógico, pero llegando únicamente hasta un cierto nivel de árbol, es decir sin alcanzar todavía las claves en las hojas del árbol. A partir de ese momento, se tiene un árbol de búsqueda parcial en el que cada nodo constituye un problema equivalente al original pero simplificado. A continuación interviene la segunda etapa del algoritmo y la computación paralela en la que cada nodo de ese nivel del árbol, se resuelve en paralelo mediante el uso de un elevado número de procesadores, es decir, aplica el mismo algoritmo de búsqueda de claves, pero ahora ya sí, hasta alcanzar las hojas del árbol, es decir, las soluciones del problema.

Existen numerosos factores a tener en cuenta a la hora de aplicar el algoritmo paralelo, de entre los cuales, el más importante es el valor de corte o parada de la primera etapa (en adelante *BOV* por sus siglas en

²<http://www.scbi.uma.es/>

inglés, *Break-Off Value*). Determinar este valor es un punto muy sensible del problema, pues de él depende el aprovechamiento general de los recursos en la aplicación del paralelismo [13]. Esto se debe a que existe la necesidad de elegir un *BOV* de forma que la primera fase del algoritmo no requiera una cantidad excesiva de tiempo de ejecución, pero al mismo tiempo, que se genere la suficiente información para poder maximizar el rendimiento de la segunda fase, la de computación paralela.

Para contrastar la aportación del algoritmo, se han realizado una considerable cantidad de pruebas de rendimiento, las cuales necesitan llevarse a cabo en entornos de supercomputación y cuyos resultados pueden consultarse en [13]. Así, se ha demostrado que el algoritmo diseñado es claramente susceptible de ejecutarse utilizando una implementación paralela. Se puede comprobar como se consiguen resultados en tiempos razonables incluso en los casos en los que la cantidad de información de entrada es considerable y en los que los métodos secuenciales no son capaces de finalizar.

1.2. Generadores Minimales

Como se ha mencionado anteriormente, una forma de representar en FCA el conocimiento es el retículo de conceptos. Esta representación otorga una visión global de la información con un formalismo muy sólido, abriendo la puerta para utilizar la teoría de retículos como una metateoría para gestionar la información [16].

Los conjuntos cerrados son la base para la generación del retículo de conceptos ya que éste puede ser construido a partir de aquellos, considerando la relación de subconjuntos como la relación de orden. En este punto nace el concepto de generadores minimales como representaciones canónicas de cada conjunto cerrado [46].

Los generadores minimales junto con los conjuntos cerrados son esenciales para obtener una representación completa del conocimiento en FCA. Su relevancia puede apreciarse a través de importantes estudios como [96, 98]. Además, los generadores minimales se han usado como punto clave para generar bases, las cuales constituyen una representación compacta del

conocimiento que facilita un mejor rendimiento de los métodos de razonamiento basados en reglas. Missaoui et al. [83,84] presentan el uso de generadores minimales para calcular bases que impliquen atributos positivos y negativos cuyas premisas son generadores minimales.

En este aspecto, el trabajo de esta parte de la tesis ha consistido en el estudio y diseño de métodos para la enumeración de todos los conjuntos cerrados y sus generadores minimales a partir del conjunto de implicaciones. El proceso se desarrolla a partir de esta información, y no del *dataset* original, lo cual, hasta donde se ha investigado, no se había hecho previamente.

Métodos para el Cálculo de Generadores Minimales

Los métodos propuestos en esta tesis son una evolución del presentado en [28], donde se utilizó la lógica SL_{FD} como medio para encontrar todos los generadores minimales a partir de un conjunto de implicaciones. Este método trabaja sobre el conjunto de implicaciones aplicando unas reglas de inferencia y construyendo árbol de búsqueda de aspecto similar a los árboles del caso de las claves minimales. No obstante, hay una diferencia esencial en el caso de los generadores minimales y es la siguiente.

Al igual que el algoritmo CK para claves minimales, los métodos propuestos utilizan el cierre SL_{FD} , con lo cual, en cada paso también obtienen un conjunto Σ' reducido de implicaciones, pero ahora además, cada nodo del tableaux que se genera es una solución parcial del problema, la cual se combinará con el resto de soluciones al término de la ejecución del algoritmo para obtener el resultado final, mientras que en el caso de claves minimales, las soluciones se encontraban únicamente en las hojas.

Tras el método presentado en [28] (que los autores denominaron MinGen) se presenta ahora un nuevo método, MinGenPr, que aplica una importante mejora con respecto al anterior. Fundamentalmente, consiste en incorporar un mecanismo de poda, basada en un test de inclusión de conjuntos, que involucra a todos los nodos del mismo nivel, para evitar la enumeración de generadores minimales y cierres redundantes. El propósito de esta poda es verificar la información de cada nodo en el espacio de

búsqueda, evitando la apertura de una rama completa.

Finalmente, se propone un último método, GenMinGen, que generaliza la estrategia de poda anterior al considerar el test de inclusión del subconjunto no sólo con la información de los nodos del mismo nivel, sino también con todos los generadores minimales calculados antes de la apertura de cada rama.

En definitiva, se han estudiado, diseñado e implementado cada uno de estos métodos en su versión secuencial. Para evaluar el rendimiento e ilustrar las mejoras obtenidas al pasar de un método a otro, se han realizado un gran número de pruebas utilizando información sintética e información real procedente de repositorios de datos utilizados comúnmente en investigación, como son los de la Universidad de California, Irvine (UCI)³.

A la luz de los resultados obtenidos en [14], se aprecia claramente como la estrategia de poda del método MinGenPr hace que su rendimiento supere con creces al anterior MinGen. Estas mejoras pueden verse reflejadas en la reducción del número de nodos del árbol de búsqueda y su consiguiente disminución de los tiempos de ejecución del algoritmo. Respecto al último método, GenMinGen, los resultados de los experimentos son aún más notables, alcanzando reducciones superiores al 75 % en ambas métricas (i.e. número de nodos y tiempos de ejecución) en muchos de los casos.

Generadores Minimales y Paralelismo

Si se pretende trabajar sobre conjuntos de implicaciones con una cantidad de información substancial, surge el mismo problema que en la enumeración de las claves minimales, la capacidad computacional de una máquina convencional actual no es suficiente para solucionar estos problemas en un tiempo razonable. Por tanto, se vuelve a utilizar el paralelismo como estrategia para abordar el problema.

Aunque GenMinGen ha demostrado tener un mejor rendimiento que MinGenPr (y ambos a su vez un mejor rendimiento que MinGen) como

³<https://archive.ics.uci.edu/ml/datasets.html>

demuestran los resultados obtenidos en [14], sólo se va a desarrollar una versión paralela del método MinGenPr, denominada *MinGenPar*. Esto se debe a que, cuando se usa el método MinGenPr, no existe necesidad de comunicación entre los nodos del árbol, y por tanto, se puede utilizar la misma filosofía paralela de implementación *MapReduce* en dos etapas que se utiliza en el caso de las claves minimales. Sin embargo, cuando se usa el método GenMinGen, es necesario comparar los resultados obtenidos en cada nodo con el conjunto actual de generadores minimales generados hasta el momento. Esto rompe esa filosofía de implementación, donde cada nodo del árbol está destinado a ser resuelto de forma independiente y sin existir comunicación entre cada uno de ellos. No obstante, esta circunstancia es el objetivo de estudio de uno de los trabajos futuros que se proponen en la Sección 6.2.

Finalmente, para verificar el rendimiento y la idoneidad de los métodos en relación a la aplicación de estrategias paralelas, se ha realizado una amplia batería de pruebas tanto sobre información sintética como información real, tal y como se ha explicado anteriormente para el tema de las claves minimales. Además, las pruebas han incluido tareas de estimación del número óptimo de cores a utilizar, así como del valor de corte más apropiado en la etapa primera de los métodos paralelos. Los resultados obtenidos respecto a esta parte de la investigación pueden consultarse en [12] y, especialmente en [14], que constituye uno de los trabajos que avalan esta tesis doctoral.

1.3. Sistemas de Recomendación Conversacionales

La tercera aportación de esta tesis doctoral haciendo uso de los conjuntos de implicaciones se enmarca en el campo de los sistemas de recomendación (SRs).

De forma muy simplificada, se podría considerar que un SR es un sistema inteligente que proporciona a los usuarios una serie de sugerencias personalizadas (recomendaciones) seleccionadas de un conjunto de elementos (ítems). Comúnmente, los SRs estudian las características de cada usuario e

ítem del sistema, y a partir de ahí, mediante un procesamiento de los datos, encuentra un subconjunto de ítems que pueden resultar de interés para el usuario. Una recopilación de las referencias más notables en el campo de los SRs la encontramos en [102].

Desde los primeros trabajos sobre SRs [56, 100], éstos han estado en continua evolución durante los últimos años [1]. Sin embargo, es con la expansión de las nuevas tecnologías cuando han tenido un acercamiento más directo a la mayor parte de la sociedad debido a su capacidad para realizar todo tipo de recomendaciones sobre productos muy populares (libros [31], documentos [97], música [68], turismo [47], películas [43, 54], etc.).

Los SRs constituyen tanto un importante campo de investigación [37, 114], como un elemento indispensable para sólidos entornos comerciales a nivel mundial (Amazon [72], LinkedIn [99], Facebook [117]), lo cual pone de manifiesto la importancia de estos sistemas en la sociedad actual.

Abordar la generación de recomendaciones haciendo uso de FCA es una aproximación existente en la literatura desde hace años. En [36], los autores utilizan FCA para agrupar elementos y usuarios en conceptos para posteriormente, realizar recomendaciones colaborativas según la afinidad con los elementos vecinos. Más tarde, en [109], se introducen un modelo para el filtrado colaborativo basado en FCA para generar correlaciones entre datos a través de un diseño del retículo. Zhang et al. [135] propusieron un sistema basado en similitud agrupando la información contextual en grafos mediante el cual llevar a cabo recomendaciones sobre las interacciones sociales entre usuarios. En [70, 71], se utilizan relaciones difusas e implicaciones ponderadas para especificar el contexto y SL_{FD} para desarrollar un proceso lineal de filtrado que permite a los SRs podar el conjunto original de elementos y así mejorar su eficiencia. Recientemente, en [139] se propone y utiliza un novedoso SR personalizado basado en el retículo de conceptos para descubrir información valiosa de acuerdo con los requisitos e intereses de los usuarios de forma rápida y eficiente. Todos estos trabajos subrayan claramente cómo FCA puede aplicarse con éxito en el campo de los SRs.

Existen numerosos tipos de SRs atendiendo a cómo se generan las re-

comendaciones. Los más extendidos son los de filtrado colaborativo [81] que basan su funcionamiento en las valoraciones de los usuarios a los elementos disponibles; y los sistemas basados en contenido [134] que proporcionan resultados que tengan características similares a otros valorados anteriormente por el usuario. Por otro lado, los SRs basados en conocimiento [77] utilizan un método de razonamiento para inferir la relación entre una necesidad y una posible recomendación.

Los SRs más importantes desde el punto de vista de esta tesis son los denominados conversacionales [50, 69]. Estos SRs se diferencian de los anteriores en el flujo de trabajo que se sigue para generar la recomendación. Este tipo de SR es la estrategia principal para el SR realizado y que ha dado lugar a una de las contribuciones que avalan esta tesis [15]. Se puede consultar una clasificación más detallada en [2, 17].

No obstante, la mejor alternativa consiste en combinar características de diferentes tipos de SRs para generar híbridos que se beneficien de las ventajas de cada uno de ellos [33]. Tal es el caso de este trabajo, en el que se ha realizado un SR híbrido que combina características de los SRs basados en conocimiento, contenido y, principalmente, conversacionales.

La evaluación de las predicciones y recomendaciones es un aspecto fundamental en los SRs [55, 101]. Los SRs requieren medidas de calidad y métricas de evaluación [51] para conocer la calidad de las técnicas, métodos y algoritmos para las predicciones y recomendaciones.

No obstante, dependiendo del SR con el que se trabaje, la evaluación se debe llevar a cabo utilizando aquellas métricas, que por su naturaleza y significado, sean coherentes con el SR que se desea evaluar. En el caso de estudio de esta tesis, dado el SR conversacional desarrollado, una medida adecuada de rendimiento consiste en calcular el número de pasos que se producen en la conversación [80]. Por contra, otras métricas tan populares como son *Precision* y *Recall* [52] no son adecuadas de aplicar en el trabajo de esta tesis porque se obtendría siempre valores máximos en ambas métricas, y la razón es la siguiente. En primer lugar, cualquier ítem de la lista de resultados, verifica los atributos seleccionados ya que la consulta para obtener los ítems resultado contiene esas restricciones. Y en segundo lugar,

a cada paso del diálogo, el sistema devuelve todos los ítems que verifiquen la selección de atributos establecida por el usuario.

Por la misma razón, no existe necesidad de considerar métricas referentes a la exactitud de los resultados ya que el sistema desarrollado no es un modelo de predicción, su funcionamiento está basado en implicaciones y eso asegura la completa de exactitud en las respuestas.

Problemas Comunes y la Maldición de la Dimensión

Si bien es cierto que los SRs están alcanzando una enorme importancia, existen numerosas dificultades que han de afrontarse a la hora de diseñarlos e implementarlos. En la lista de problemas relacionados con los SRs [110] se pueden destacar: el arranque en frío [42, 115], privacidad [44], oveja-negra [49], escasez [53], ataques maliciosos [130, 137], sobreespecialización [73], escalabilidad [59], postergación [116], dimensionalidad [107], etc.

En concreto, en esta tesis se ha orientado el trabajo a abordar este último problema de la dimensionalidad en los SRs. Este problema, también conocido como *the curse of dimensionality phenomenon* [89, 107] aparece cuando es necesario trabajar sobre *datasets* con un alto número de características (variables o atributos). De forma intuitiva, se puede describir de la siguiente manera: cuando hay pocas columnas de datos, los algoritmos de tratamiento inteligente de la información (aprendizaje automático, *clustering*, clasificación, etc.) suelen tener un buen comportamiento. Sin embargo, a medida que aumentan las columnas o características de nuestros ítems, se vuelve más difícil hacer labores predictivas con un buen nivel de precisión. El número de filas de datos necesarias para realizar cualquier modelado útil aumenta exponencialmente a medida que agregamos más columnas a una tabla [79].

Para abordar este problema, se pueden encontrar numerosos trabajos en la literatura [66, 107], especialmente mediante selección de características, que pueden ayudar a descartar aquellas características que no son relevantes

de cara al objetivo buscado. De hecho, estas técnicas ya se aplican en otras áreas como son: algoritmos genéticos o redes neuronales, normalmente centrándose en la aplicación de un proceso automatizado por lotes [123].

Un trabajo interesante en esta área es [60], que establece la idoneidad de los enfoques basados en el conocimiento para los procesos conversacionales. En particular, estos autores utilizan el razonamiento basado en restricciones, en lugar de nuestro enfoque basado en la lógica. Además, este trabajo trata sobre concepto de optimización de consultas, análogo al aplicado en la propuesta de esta tesis. Otro trabajo notable es [118], que comparte el objetivo de disminuir el número de pasos de la conversación. Los autores proponen métricas acerca del número de pasos y tasas de poda, ambas muy similares a las utilizados en este trabajo de tesis. Por otro lado, en [20], los autores demuestran cómo la posibilidad de que sea el usuario el encargado de la selección de atributos genera una ventaja con respecto al hecho de que sea el sistema mismo el encargado de dicha selección. Este hecho respalda el enfoque buscado en esta tesis, en el cual el experto humano guía la conversación y el proceso de selección de características.

Propuesta Desarrollada

El objetivo ha sido abordar el problema de la alta dimensionalidad en los SRs haciendo uso de los conjuntos de implicaciones, a través de un proceso de selección de atributos por parte del usuario mediante el SR híbrido mencionado anteriormente. De esta forma, se ha conseguido reducir el número de pasos necesarios en el diálogo y gestionar favorablemente el problema de la dimensionalidad [15].

Así, el sistema desarrollado se va a centrar principalmente en la primera fase de la recomendación, el filtrado. Para ello, partiendo del conjunto de implicaciones, utiliza la lógica SL_{FD} [26] y, especialmente, el algoritmo del cierre SL_{FD} [87] como motor para facilitar y acelerar la recomendación. Gracias a la aplicación del cierre, el sistema reduce la sobrecarga de información a cada paso del diálogo filtrando aquellos atributos que resulten de la aplicación del cierre a las solicitudes del usuario, consiguiendo una

reducción del número de pasos necesarios en el diálogo.

Se han realizado numerosas pruebas de aplicación para contrastar su validez. Entre ellas, destacan las pruebas utilizando información real sobre enfermedades y fenotipos, como se puede apreciar en una de las contribuciones que avalan este trabajo de investigación [15]. Además, la propuesta desarrollada, al igual que la gran mayoría de los SRs, casa con los conceptos de adaptabilidad y longevidad de los SRs ya que el funcionamiento es independiente de la información de base con la que trabaje, sólo es necesario conocer el conjunto de atributos e implicaciones subyacente a los datos.

1.4. Verificación de los Resultados

Antes de entrar plenamente en el cuerpo de la tesis es necesario hacer una importante aclaración previa con la intención de indicar la manera de certificar la validez de los resultados obtenidos a lo largo de la tesis.

Como se verá en los siguientes capítulos, la labor de investigación se ha centrado en actuar sobre conjuntos de implicaciones, y en ese sentido, para los experimentos realizados se ha contado con unos ficheros de entrada que contenían la información necesaria, y sobre ellos se han obtenido unos resultados. Ahora bien, la forma de verificar que esos resultados son correctos es la siguiente.

En primer lugar y con respecto a los resultados que se presentan en cuanto a claves y generadores minimales, se han realizado numerosos ejercicios en papel intentando buscar casos límites donde la implementación pudiera no ser precisa y se ha comprobado que los resultados obtenidos en papel coincidían exactamente con los calculados por la máquina. Además, dado que para muchos de los experimentos, en los que se llegaban a calcular millones de nodos de un árbol, no era posible comprobar si cada uno de esos cálculos era correcto, para el caso concreto de los experimentos relacionados con claves minimales, la validez de los experimentos viene dada al haber cotejado los resultados con aquellos obtenidos sobre un amplio abanico de experimentos en trabajos anteriores [9, 10] donde su validez quedó demostrada. Además, la validez de los resultados se corrobora igualmente al

alcanzar las mismas soluciones para diferentes métodos cuando cada uno de ellos hace un tratamiento de la información diferente con respecto al otro. En relación a los experimentos con SRs conversacionales, dado que los experimentos no alcanzan números tan costosos de verificar, la validez de los resultados puede demostrarse de forma más asequible siguiendo un desarrollo explícito en papel.

Adicionalmente, y con mayor énfasis en relación a los experimentos que han conllevado la utilización de recursos de supercomputación, cada uno de los experimentos se ha reproducido entre 30 y 50 veces, de forma que los resultados mostrados son fruto de un estudio estadístico posterior más amplio que permite identificar los resultados más fiables, tal y como se sugiere en [48, 129, 138].

Para cada una de las implementaciones realizadas, existía la necesidad de establecer criterios que permitieran evaluar el rendimiento de las pruebas de forma que se pudieran comparar unos métodos con otros. En el caso de los experimentos con claves y generadores minimales, cuando se plantea la idea de la comparación de resultados, lo primero que se pensó fue la medición de los tiempos que necesitaba cada uno de los métodos para obtener los resultados. No obstante, se advirtió que este parámetro está íntimamente ligado a la arquitectura que estamos utilizando para ejecutar el experimento, lo cual hace que el resultado dependa en gran medida de los recursos que se están utilizando y no tanto de la calidad o eficiencia del propio algoritmo. En consecuencia, se oscurecía la utilidad teórica de los resultados obtenidos. Por tanto, se decidió contabilizar la magnitud del árbol y la cantidad de resultados redundantes que se obtienen (véase [9–11]). De esta forma, en el momento de que exista otro método con un código en cualquier otro lenguaje o utilizase recursos *hardware* diferentes que desembocaran en una mejora del tiempo, siempre se puede atender al tamaño del árbol y al número de cálculos redundantes, pudiendo defender si realmente es una mejora en el método o bien, en la ejecución debido a la arquitectura.

A modo de resumen gráfico, la Figura 1.1 muestra un esquema del camino de investigación que se ha seguido en el desarrollo de esta tesis doctoral, apoyado por las principales nociones y referencias.

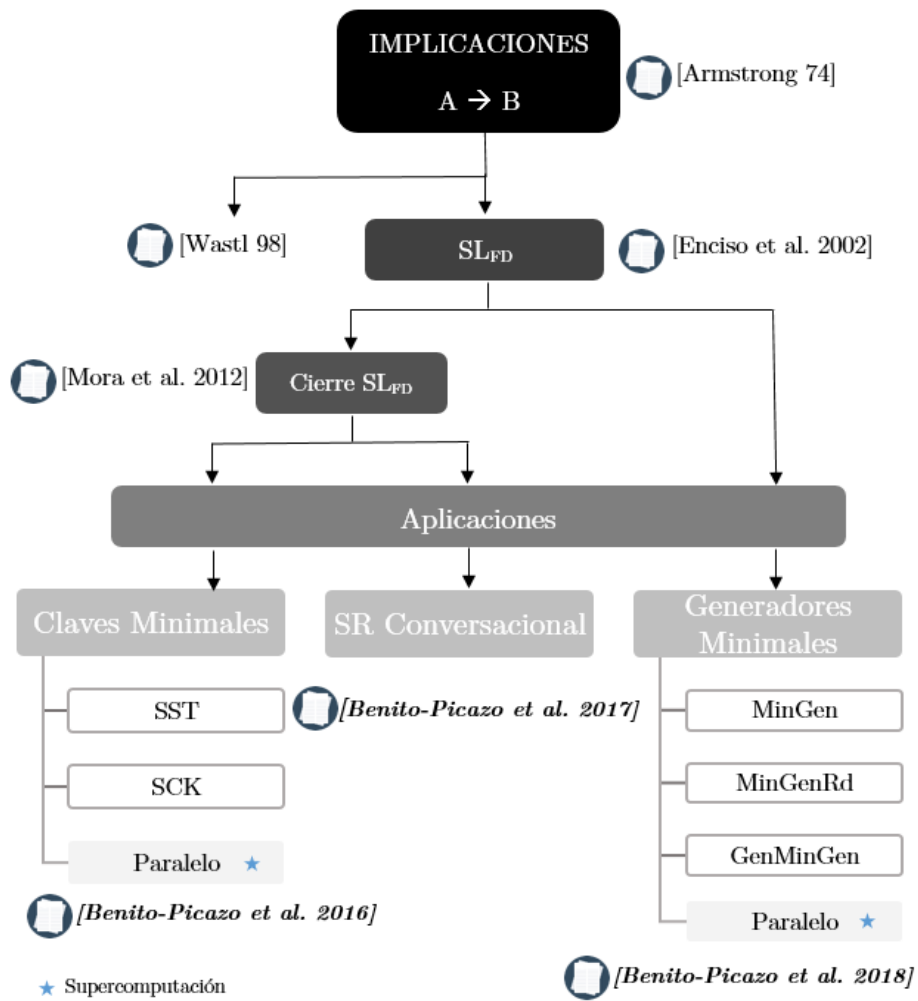


Figura 1.1: Esquema del estado del arte y las contribuciones generadas.

Para finalizar este capítulo, se incluye un último apartado donde se describe brevemente la estructura que presenta el documento incluyendo las publicaciones que avalan esta tesis.

1.5. Estructura de la Tesis

En este primer capítulo de introducción, se han fijado los puntos fundamentales de la tesis, como son: el marco de trabajo sobre el que se va a actuar, las técnicas que se utilizarán y los principales objetivos que se pretenden alcanzar. Concretamente, se ha estipulado la utilización de los conjuntos de implicaciones y las DFs como base del estudio sobre la que aplicar técnicas basadas en la lógica para mejorar el tratamiento de la información.

Tras la introducción, aparece el Capítulo 2, en el que se presentan: el conjunto de nociones principales de FCA para el trabajo realizado sobre generadores minimales y SRs, los aspectos fundamentales de bases de datos y DFs para la investigación sobre claves minimales, las lógicas de implicaciones y los métodos de razonamiento automático utilizados.

A continuación, se presenta el Capítulo 3 en el que se presenta la primera contribución que avala este trabajo de investigación y que corresponde con el trabajo realizado en el campo de la búsqueda de las claves minimales. De forma general, este artículo presenta nuevos métodos para resolver el problema de la inferencia de claves minimales en esquema de datos basándose en la lógica SL_{FD} y el uso de implicaciones. Además, se muestra el funcionamiento de esos métodos y las ventajas obtenidas al aplicar técnicas de computación paralela para poder aplicar los métodos sobre conjuntos de información de un tamaño tal que las técnicas secuenciales no son capaces de gestionar en cuanto a tiempo y recursos necesarios.

Seguidamente, se presenta el Capítulo 4, análogo al anterior pero esta vez para el tema referente a los generadores minimales 3. Este capítulo presenta un segundo artículo en el cual se lleva a cabo un estudio de los métodos de producción de generadores minimales basados en la lógica y el tratamiento de implicaciones. Se comprueba las mejoras de rendimiento de los métodos al aplicar reducciones en el espacio de búsqueda basadas

en estrategias de poda. Al igual que en el caso de las claves minimales, se presenta el funcionamiento de los métodos paralelos para poder tratar con conjuntos de información de tamaño considerable y se incluyen las pruebas realizadas en entornos de supercomputación.

Como último capítulo dedicado a las aplicaciones desarrolladas mediante la gestión de implicaciones se incluye el Capítulo 5. En este capítulo se presenta un novedoso trabajo en el que se desarrolla una aproximación al tratamiento del problema de la dimensionalidad en los SR. Mediante el uso de las implicaciones, la lógica SL_{FD} y el algoritmo del cierre SL_{FD} , se desarrolla un modelo de SR conversacional. Este sistema, es capaz de gestionar el problema de la dimensionalidad reduciendo la sobrecarga de información con la que el usuario debe enfrentarse a la hora de obtener una recomendación. Esta reducción se consigue mediante un filtrado de atributos guiado por la aplicación del cierre. Se demuestra su buen comportamiento mediante su evaluación sobre información real.

Finalmente, la tesis se cierra con el Capítulo 6 dedicado a recopilar las principales conclusiones obtenidas y a proponer caminos por los que seguir ahondando en la investigación. Además, se incluye una relación de las referencias consultadas y los respectivos índices de términos, figuras y tablas.

En aras de la completitud, se incluyen como anexos finales aquellos artículos que han sido publicados a lo largo de este periodo de investigación, que si bien no se utilizan como respaldo para esta tesis doctoral, han sido la semilla y experiencia inicial a partir de la cual se han desarrollado los trabajos que actúan como aval. En la Figura 1.2 se muestra de forma gráfica el contenido de la tesis y se contextualizan las contribuciones publicadas.

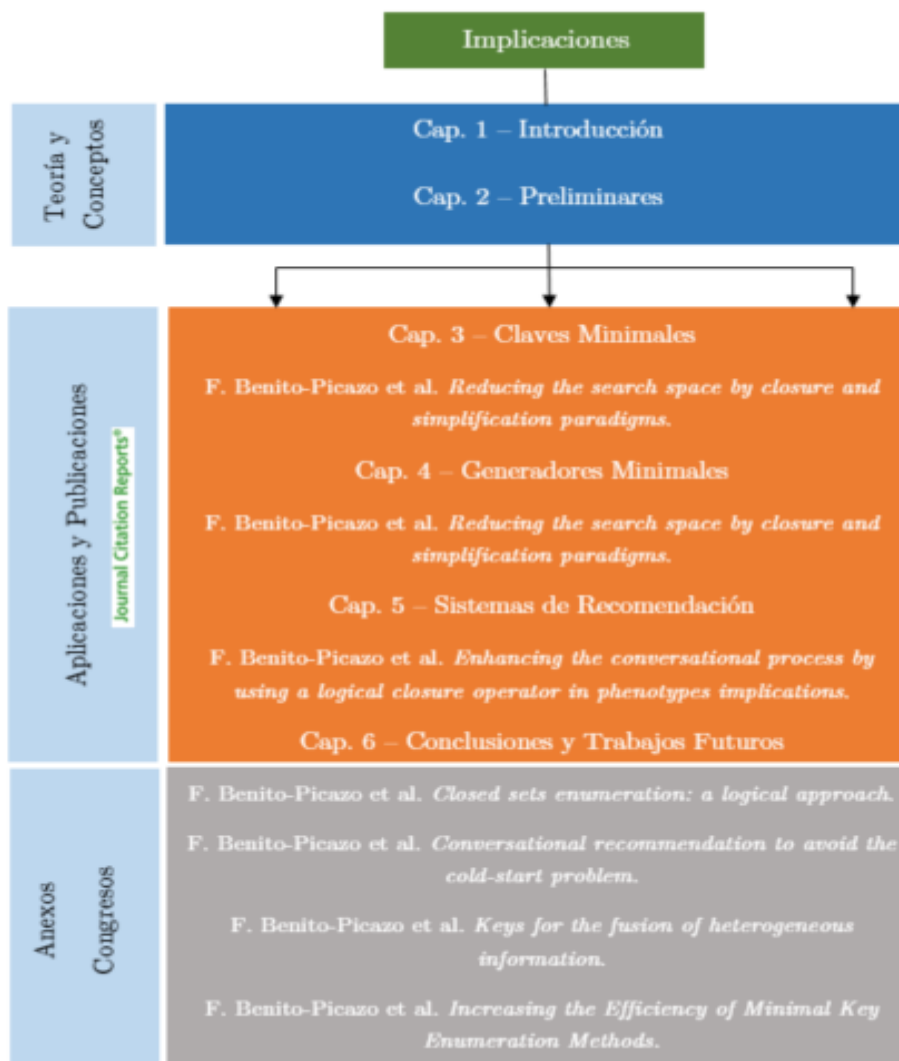


Figura 1.2: Esquema de la estructura de la tesis y las publicaciones.



UNIVERSIDAD
DE MÁLAGA

Capítulo 2

Preliminares



UNIVERSIDAD
DE MÁLAGA

*Todo encajaba, con la abrumadora evidencia
de una demostración matemática.*

Ethan de Athos

L. M. Bujold

A lo largo de este capítulo se van a introducir los principales conceptos, definiciones y resultados sobre los que se sustentan el trabajo de investigación desarrollado. Se parte una vez más de que el objetivo principal de la tesis doctoral es la utilización de los conjuntos de implicaciones para generar soluciones a problemas en el ámbito de FCA, bases de datos y sistemas de recomendación. Se ha diseñado este capítulo con la intención de que el texto sea autocontenido en la medida de lo posible y para ello se ha dividido en cuatro secciones principales tal como muestra la Figura 2.1.

La primera sección presenta las nociones de FCA que serán necesarias para las soluciones propuestas para generadores minimales y sistemas de recomendación. En la segunda, se recopilan los aspectos fundamentales a tener en cuenta para la enumeración de las claves minimales en esquemas relacionales por medio de DFs. La tercera parte se dedica a presentar formalmente las lógicas de implicaciones. La cuarta y última, se centra en el razonamiento automático usando dichas lógicas.

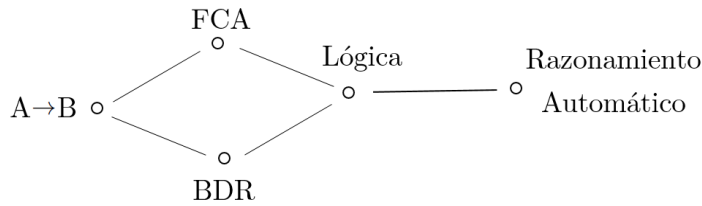


Figura 2.1: Esquema de contenido del Capítulo 2, Preliminares.

2.1. Análisis Formal de Conceptos

De forma general, se puede considerar FCA como un marco para el análisis de información que facilita la extracción de conocimiento y la posibilidad de poder razonar sobre él. La referencia principal de este campo de conocimiento viene de la mano de Wille y Ganter en [46]. Según el propio Wille en [127]: “*El objetivo y significado del análisis formal de conceptos como teoría matemática consiste en apoyar la comunicación racional de las personas, mediante el desarrollo de estructuras de conceptos matemáticamente apropiadas que puedan activarse lógicamente*”.

Como principales fortalezas de FCA cabe mencionar: su sólida base matemática y filosófica, una representación gráfica e intuitiva del conocimiento, avalada en más de 2.000 publicaciones científicas, y aplicaciones en cientos de proyectos [127] en diversos campos de conocimiento, como se ha mencionado en el Capítulo 1.

2.1.1. Contextos Formales

El punto de partida de FCA es un *contexto formal*. Es una noción fundamental para el contenido de esta sección y se define a continuación.

Definición 2.1.1 (Contexto formal). *Un contexto formal es una tripleta $K = (G, M, I)$ que consiste en dos conjuntos no vacíos, G y M , y una relación binaria I entre ellos. Los elementos de G se llaman objetos del contexto, y los elementos de M se llaman atributos del contexto. Para $g \in G$*

y $m \in M$, escribimos $\langle g, m \rangle \in I$ o gIm si el objeto g posee el atributo m .

La forma más sencilla de representar un contexto formal es mediante una tabla donde los objetos se sitúan en sus filas y los atributos en las columnas, de forma que un valor en cada celda indica si un objeto $g \in G$ posee un atributo $m \in M$. Hay que tener en cuenta que la definición de contexto formal es deliberadamente muy general. No hay restricciones sobre la naturaleza de los objetos y atributos. Se pueden considerar objetos físicos, personas, números, procesos, estructuras, etc. En realidad, cualquier cosa que sea un *conjunto* en el sentido matemático se puede tomar como el conjunto de objetos o de atributos de algún contexto formal. También es posible intercambiar el papel de los objetos y atributos, es decir, si se tiene que $K = (G, M, I)$ es un contexto formal, igualmente lo será su dual, $K = (M, G, I')$ con $mI'g \iff gIm$. Además, ni siquiera es necesario que G y M sean disjuntos, de hecho, pueden incluso no ser diferentes.

Para ilustrar un ejemplo de contexto formal y como apoyo al contenido que se presenta en esta sección, se va a utilizar el siguiente ejemplo tomado de [45]. En él, los autores muestran un información sobre los destinos aéreos que oferta un grupo de aerolíneas comerciales.

Ejemplo 2.1.2. Sea K un contexto formal donde el conjunto de objetos G comprende todas las líneas aéreas del grupo *Star Alliance* y el conjunto de atributos M muestra sus destinos.

La relación binaria I viene reflejada en la Tabla 2.1 y muestra los destinos a los que viaja cada miembro de *Star Alliance*. Por tanto, se tiene el contexto formal $K = (G, M, I)$ en el cual:

$$G = \{Air\ Canada, Air\ New\ Zealand, All\ Nippon\ Airways, Ansett\ Australia, The\ Austrian\ Airlines\ Group, British\ Midland, Lufthansa, Mexicana, Scandinavian\ Airlines, Singapore\ Airlines, Thai\ Airways\ International, Unites\ Airlines, VARIG\}$$

$$M = \{Latinoamérica, Europa, Canadá, Asia, Oriente\ Medio, África, México, Caribe, Estados\ Unidos\}$$

Tabla 2.1: Ejemplo de contexto formal sobre los destinos aéreos del grupo Star Alliance [45]

	Latinoamérica	Europa	Canadá	Asia	Oriente Medio	África	México	Caribe	Estados Unidos
Air Canada	✓	✓	✓	✓	✓		✓	✓	✓
Air New Zealand		✓		✓					✓
All Nippon Airways		✓		✓					✓
Ansett Australia				✓					
The Austrian Airlines Group		✓	✓	✓	✓	✓			✓
British Midland		✓							
Lufthansa	✓	✓	✓	✓	✓	✓	✓		✓
Mexicana	✓		✓				✓	✓	✓
Scandinavian Airlines	✓	✓		✓		✓			✓
Singapore Airlines		✓	✓	✓	✓	✓			✓
Thai Airways International	✓	✓		✓				✓	✓
Unites Airlines	✓	✓	✓	✓			✓	✓	✓
VARIG	✓	✓		✓		✓	✓		✓

2.1.2. Operadores de Derivación

Dada una selección $A \subseteq G$ de objetos de un contexto formal $K = (G, M, I)$, se desea caracterizar qué atributos de M son comunes a todos estos objetos. Esto define un operador que produce para cada conjunto de objetos $A \subseteq G$, el conjunto A' de sus atributos comunes, y dualmente, a partir de un conjunto de atributos, caracterizar el conjunto de atributos que tienen en común a dichos atributos. Estos operadores se denominan operadores de derivación para K .

Definición 2.1.3 (Operadores de derivación). *Dado un contexto formal $K = (G, M, I)$, se definen los operadores de derivación:*

$$\begin{aligned}
 ()' : 2^G &\rightarrow 2^M & ()' : 2^M &\rightarrow 2^G \\
 A' &= \{m \in M \mid g I m \quad \forall g \in A\} & B' &= \{g \in G \mid g I m \quad \forall m \in B\}
 \end{aligned}$$

Ambas funciones se denotan con el mismo símbolo porque no hay lugar

a confusión.

Ejemplo 2.1.4. *A partir del contexto formal representado en la Tabla 2.1 se pueden aplicar los operadores de derivación y obtener, por ejemplo:*

- $\{Mexicana\}' = \{Latinoamerica, Canada, Mexico, Caribe, Estados Unidos\}$
- $\{Latinoamerica, Europa, Asia, Africa, Estados Unidos\}' = \{Scandinavian Airlines, Lufthansa, VARIG\}$

Si A es un conjunto de objetos, entonces A' es un conjunto de atributos, al cual podemos aplicar el segundo operador de derivación para obtener el conjunto de objetos A'' . De forma dual, comenzando con un conjunto B de atributos, podemos formar el conjunto de atributos B'' .

De la definición anterior se obtiene que:

El operador de cierre $''$ (i.e. aplicar el operador de derivación $'$ y su dual) verifica algunas propiedades interesantes que serán fundamentales para poder desarrollar la teoría formal.

Definición 2.1.5 (Operador de cierre). *Sea $K = (G, M, I)$ un contexto formal, entonces el operador $()'' : 2^M \rightarrow 2^M$ es un operador de cierre, es decir, satisface las siguientes propiedades:*

- *Idempotente:* $X'''' = X'' \quad \forall X \in 2^M$
- *Monótona:* $X \subseteq Y \rightarrow X'' \subseteq Y'' \quad \forall X, Y \in 2^M$
- *Extensiva:* $X \subseteq X'' \quad \forall X \in 2^M$

Por dualidad, el resultado es igualmente válido para $()'' : 2^G \rightarrow 2^G$.

Un conjunto $A \subseteq M$ se denomina *conjunto cerrado* para el operador $''$ si es un punto fijo para $''$, es decir, $A'' = A$. Los conjuntos cerrados permiten definir lo que se denominan *conceptos formales*, los cuales son una noción esencial en FCA y a los que se dedica el siguiente apartado.

Ejemplo 2.1.6. *A partir del contexto formal representado en la Tabla 2.1 se puede obtener el conjunto cerrado de atributos:*

$Z = \{\text{Latinoamerica, Europa, Canada, Asia, Oriente Medio, Africa, Mexico, Estados Unidos}\}.$

Ya que $Z'' = Z$.

2.1.3. Conceptos Formales

De forma general, un concepto formal permite describir formalmente un hecho del contexto y caracterizar un conjunto de objetos por medio de los atributos que comparten y viceversa.

Así, un par (X, Y) con $X \subseteq G$ y $Y \subseteq M$ es un concepto formal cuando se verifica que:

- Cada objeto en X tiene todos los atributos de Y , y dualmente, cada atributo de Y está presente en cada objeto de X .
- Para cada objeto en G que no está en X , existe un atributo en Y que el objeto no tiene. Dualmente, para cada atributo en M que no está en Y , hay un objeto en X que no tiene ese atributo.

Se introduce ahora la noción formalmente:

Definición 2.1.7 (Concepto formal). *Sea $K = (G, M, I)$ un contexto formal y $A \subseteq G$, $B \subseteq M$. El par (A, B) se denomina concepto formal si $A' = B$ y $B' = A$. El conjunto de objetos A se denomina extensión del concepto (A, B) mientras que el conjunto de atributos B será la intensión del concepto.*

La descripción de un concepto a través de su extensión e intensión es redundante, ya que cada una de las dos partes determina la otra debido a que $B = A'$ y $A = B'$, sin embargo, pueden existir ocasiones en las cuales esta descripción redundante puede ser conveniente [45].

Una alternativa gráfica de identificar los conceptos formales es la siguiente. A partir de la representación del contexto formal, un concepto formal se puede reconocer por medio de un conjunto de elementos que

comparten exactamente las mismas relaciones. De esta forma, en la Tabla 2.2, se ve como, a partir del contexto formal, es posible identificar el concepto formal: $\{Air\ New\ Zealand, All\ Nippon\ Airways\}, \{Europa, Asia, Estados\ Unidos\}$.

Tabla 2.2: Extracto del ejemplo de contexto formal sobre los destinos aéreos del grupo Star Alliance 2.1

	Latinoamérica	Europa	Canadá	Asia	Oriente Medio	África	México	Caribe	Estados Unidos
[...]									
Air New Zealand		✓		✓					✓
All Nippon Airways		✓		✓					✓
[...]									

2.1.4. Retículo de Conceptos

Un contexto formal puede tener muchos conceptos formales. El conjunto de todos los conceptos formales de un contexto formal K tiene estructura de *retículo* con la relación de orden que se muestra a continuación.

Si (X_1, Y_1) y (X_2, Y_2) son conceptos, se define un orden parcial, \leq , de forma que $(X_1, Y_1) \leq (X_2, Y_2)$ si y sólo si $X_1 \subseteq X_2$, o equivalentemente, si $Y_2 \subseteq Y_1$.

La Figura 2.2 muestra el retículo de conceptos asociado al contexto formal mostrado anteriormente en el Ejemplo 2.1.2. En un diagrama como el de la Figura 2.2, cada nodo representa un concepto formal. Un concepto c_1 es un subconcepto de un concepto c_2 si y sólo si hay un camino descendente desde el nodo que representa c_2 al nodo que representa c_1 . El nombre de un objeto g se asocia al nodo que representa el concepto más pequeño que contiene g en su extensión; dualmente, el nombre de un atributo m va asociado al nodo que representa el concepto más grande con m en su extensión.

Se pueden comprobar fácilmente las relaciones que existen en el contexto ya que un objeto g tiene un atributo m si y sólo si el concepto asociado a g es un *subconcepto* del asociado a m . La extensión de un concepto consiste en todos aquellos objetos cuyas etiquetas están asociadas a subconceptos, y, dualmente, la intensión consiste en todos los atributos asociados a *superconceptos*.

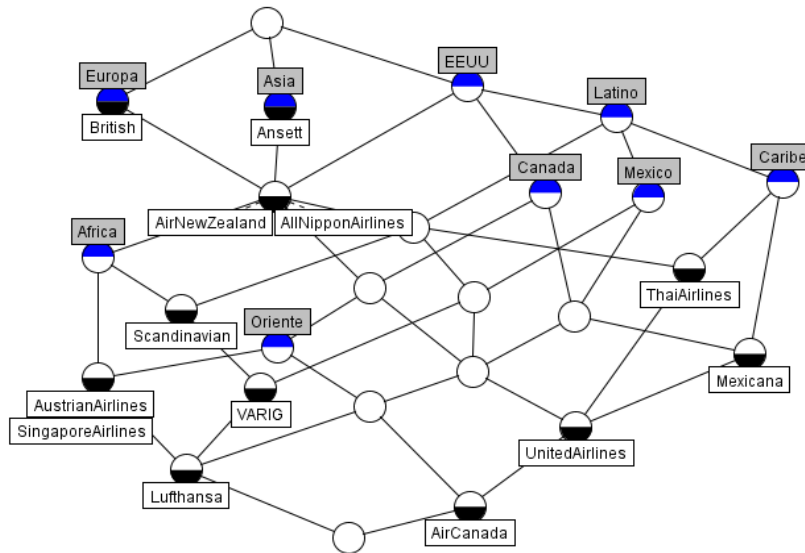


Figura 2.2: Retículo de conceptos asociado al contexto formal 2.1.2

Dicho lo anterior y teniendo en cuenta la sobrecarga que conllevaría representar cada concepto del retículo en la Figura 2.2, se puede mencionar aquí como ejemplo, el concepto etiquetado como ‘Oriente’ (*Oriente Medio*), el cual tiene como extensión el conjunto $\{Singapore Airlines, The Austrian Airlines Group, Lufthansa, Air Canada\}$, y como intensión $\{Oriente Medio, Canadá, Estados Unidos, Europa, Asia\}$.

En la parte superior del retículo, encontramos los destinos que ofrecen

la mayoría de las aerolíneas: Europa, Asia Pacífico y los Estados Unidos. Por ejemplo, exceptuando British Midland y Ansett Australia, todas las aerolíneas ofrecen vuelos a Estados Unidos. Esas dos líneas aéreas se encuentran en la parte superior del diagrama, ya que tienen la menor oferta de vuelos, operan sólo en Europa y Asia, respectivamente.

De esta forma, cuanto más se descienda en el retículo de conceptos, más destinos ofrecerán las aerolíneas. Así, la mayor oferta de destinos se encuentra en las aerolíneas de la parte inferior del retículo: Lufthansa y Air Canada. De forma análoga, conforme más se descienda en el retículo, se encontrarán los destinos menos ofertados, e.g. África, Oriente Medio y el Caribe.

2.1.5. Sistema de Implicaciones

Una noción equivalente al retículo de conceptos es el denominado sistema de implicaciones [46]. Como ya se adelantó en el Capítulo 1, las implicaciones constituyen el eje fundamental de trabajo en esta tesis doctoral.

Para introducir la labor realizada utilizando la lógica sobre los conjuntos de implicaciones, se va a comenzar describiendo los componentes habituales de una lógica: su lenguaje, semántica, sistema axiomático y método de razonamiento automático. No obstante, como se ha comentado anteriormente, los conjuntos de implicaciones se han utilizado para diferentes campos de conocimiento (FCA, bases de datos). En consecuencia, en este punto se presenta el elemento de la lógica común a todos ellos, en concreto, el lenguaje. El resto se presentará para cada una de las secciones correspondientes 2.2 y 2.3.

Lenguaje

Definición 2.1.8. *Dado un conjunto M finito de símbolos (denominados atributos) no vacío, el lenguaje sobre M se define como:*

$$\mathcal{L}_M = \{A \rightarrow B \mid A, B \subseteq M\}$$

Las fórmulas $A \rightarrow B$ se denominan *implicaciones* y los conjuntos A y B reciben el nombre de *premisa* y *conclusión* de la implicación respectivamente. Los conjuntos $\Sigma \subseteq \mathcal{L}_M$ se denominan *conjuntos de implicaciones* sobre M .

Se utilizará la siguiente notación:

- Se utilizarán letras minúsculas para denotar los elementos en M , mientras que las mayúsculas denotan sus subconjuntos.
- Se omiten las llaves en premisas y conclusiones, es decir, $abcde$ denota el conjunto $\{a, b, c, d, e\}$.
- Se escriben los elementos de 2^M por yuxtaposición, es decir, para $X \cup Y$ se escribirá XY .
- Para la diferencia se utiliza $X \setminus Y$.

Semántica

Tras la definición del lenguaje se pasa ahora a introducir la semántica utilizada, para lo cual se va a utilizar la noción de *operador de cierre* definido anteriormente.

Definición 2.1.9 (Modelo). *Sea $K = (G, M, I)$ un contexto formal y sea $A \rightarrow B \in \mathcal{L}_M$. El contexto K es un modelo para $A \rightarrow B$ si $B \subseteq A''$. Se denota por $K \models A \rightarrow B$.*

La noción de modelo puede extenderse a los sistemas de implicaciones de la siguiente manera. Dado $\Sigma \subseteq \mathcal{L}_M$, la expresión $K \models \Sigma$ indica que $\Sigma \models A \rightarrow B$ para toda $A \rightarrow B \in \Sigma$.

Ejemplo 2.1.10. *De la información mostrada en el contexto formal del Ejemplo 2.1.2 se puede identificar que: $K \models \text{Caribe} \rightarrow \text{Latinoamerica}$. Y también que: $K \not\models \text{Caribe} \rightarrow \text{Mexico}$.*

Al comenzar el capítulo se habló de la equivalencia entre la información que muestra el retículo de conceptos y las implicaciones; tras haber

introducido formalmente el lenguaje y la semántica de las implicaciones, ahora ya se pueden mencionar algunos ejemplos que reflejen esta equivalencia utilizando la información del retículo de la Figura 2.2. Así, es posible identificar ejemplos de implicaciones tales como:

- $Canada \rightarrow Estados Unidos$, ya que se puede ver como el primer atributo aparece por primera vez en la jerarquía en un subconcepto del nodo que refleja la primera aparición del segundo. En otras palabras, todo objeto g del contexto que tenga el atributo $Canada$ tendrá el atributo $Estados Unidos$.

Se puede considerar a las implicaciones y el retículo de conceptos como distintas alternativas para tratar la información que se puede extraer de un contexto formal. No obstante, los retículos de conceptos permiten una representación gráfica, mientras que la diferencia fundamental, desde el punto de vista de esta tesis doctoral, es que los sistemas de implicaciones proporcionan una manipulación simbólica usando la lógica de implicaciones, y por tanto, permiten el razonamiento automático.

Como se ha mencionado anteriormente, se entra ahora en la segunda sección de este capítulo en la cual se van a utilizar las implicaciones en el ámbito de las bases de datos relacionales, donde recibirán el nombre de Dependencias Funcionales.

2.2. Bases de Datos Relacionales

El modelo de base de datos relacional aparece en el reconocido artículo de Edgar Frank Codd en 1970 [21]. Codd propuso que los sistemas de bases de datos deberían presentarse al usuario mediante una vista de datos organizada en forma de tablas bidimensionales (filas y columnas) que denominó *relaciones*.

Formalmente, en el modelo relacional, una base de datos consiste en una o más relaciones [38]. Una relación R definida sobre un conjunto de dominios D_1, D_2, \dots, D_n está formada por un esquema y un cuerpo como se definen a continuación.

Definición 2.2.1 (Esquema). *Un esquema E para una relación R se define como un conjunto fijo de pares (atributo:dominio). Habitualmente se denota como $\{(A_1 : D_1), (A_2 : D_2), (A_n : D_n)\}$ donde cada A_j corresponde a un único D_j y los A_j son todos distintos.*

En el Ejemplo 1.1.1, el esquema sería:

{
Título:Cadena de caracteres, Año:Número natural, País:Cadena de caracteres, Director:Cadena de caracteres, Nacionalidad:Cadena de caracteres, Actor:Cadena de caracteres
 }.

El conjunto de los esquemas para las relaciones de una base de datos se denomina *esquema de base de datos relacional*.

Definición 2.2.2 (Cuerpo). *Un cuerpo C para una relación R se define como un conjunto de tuplas de pares (atributo:valor). Habitualmente se denota como $\{(A_1 : v_{i1}), (A_2 : v_{i2}), (A_n : v_{in})\}$ con $i = 1, 2, \dots, m$ tal que m es el cardinal de R , i.e. el número de tuplas de la relación. En cada $(A_j : v_{ij})$ se tiene que $v_{ij} \in D_j$.*

En el Ejemplo 1.1.1, un extracto del cuerpo sería:

{
 [(*Título:Pulp Fiction*), (*Año:1994*), (*País:USA*), (*Director:Quentin Tarantino*), (*Nacionalidad:USA*), (*Actor:Uma Thurman*)],
 [(*Título:King Kong*), (*Año:2005*), (*País:NZ*), (*Director:Peter Jackson*), (*Nacionalidad:NZ*), (*Actor:Naomi Watts*)],
 ...
 }.

Tras haber definido formalmente el cuerpo y el esquema de un relación, se puede mencionar que en los sistemas gestores de bases de datos relacionales, las relaciones se representan como una tabla de doble entrada, en la que (con un una terminología más informal) se encuentran los siguientes elementos:

- **Atributo.** Cada una de las columnas de una relación se identifica con un atributo.
- **Tupla.** Se identifica cada tupla de una relación con una fila de la tabla, exceptuando la fila que contiene a los atributos.

2.2.1. Dependencias Funcionales

Un diseño defectuoso del esquema de base de datos relacional puede provocar lo que se denominan *anomalías* [61, 108, 120]. Estas anomalías son problemas que aparecen a la hora de operar con la base de datos en términos de actualizaciones, eliminaciones o inserción de información redundante [30, 121].

La manera de eliminar estas anomalías consiste en descomponer las relaciones [5]. La descomposición de relaciones conduce al campo de la *normalización* [23] que si bien no forma parte del ámbito principal de esta tesis doctoral, permite dar paso a la introducción del concepto de DF, el cual nace como un recurso para caracterizar la semántica de las relaciones y, por consiguiente, permitir hacer una aplicación de las transformaciones de normalización [32]. Ésta es una de las diferencias del tratamiento de las relaciones entre FCA y bases de datos relacionales, ya que para el primero, lo que se pretende es capturar conocimiento, mientras que en el segundo, las DFs se utilizan para definir formas normales [5, 41, 64].

Teniendo en cuenta que la sintaxis que se va a utilizar para DFs es la misma que la ya introducida para las implicaciones en FCA en la Sección 2.1.5, donde el conjunto M lo forman los atributos del esquema de la relación, se pasa ahora a describir la semántica para las DFs.

Semántica

Definición 2.2.3 (Dependencia Funcional). *Una dependencia funcional $X \rightarrow Y$ se cumple en una tabla R si y sólo si para cada dos tuplas de R , si sus valores en X coinciden, entonces también coinciden sus valores en Y .*

Intuitivamente, una DF $X \rightarrow Y$ indica que el conjunto de atributos X determina el conjunto de atributos Y , lo cual refleja fielmente la noción de función (y de ahí su nombre) entre los dominios de A y B , $f : A \rightarrow B$. Estas funciones son las que se usan para tratar las anomalías mencionadas anteriormente, buscando aquellas funciones que se correspondan con relaciones de forma unívoca. El siguiente ejemplo ilustra cómo la semántica de la DFs y las implicaciones en FCA es diferente.

Ejemplo 2.2.4. *Sea el contexto formal $K = (G, M, I)$ representado a continuación:*

	a	b	c
g_1	1	0	1
g_2	1	1	1
g_3	0	0	0
g_4	0	1	1

Dado ese contexto formal, se puede ver que se verifica la implicación $a \rightarrow c$ en K , mientras que por otro lado, al considerar el contexto formal como una relación del modelo relacional y considerando la semántica de DF, no se cumple que $a \rightarrow c$ debido a los diferentes valores de g_3 y g_4 en el atributo c .

La forma en que las DFs se asemejan a las implicaciones no viene dada porque se parezcan en su semántica sino por el hecho de que se pueden manejar utilizando la misma lógica, los axiomas de Armstrong [4], que se presentarán más adelante en la Sección 2.3.1.

La diferencia principal en la interpretación de las DFs y las implicaciones proviene de la estructura de los *datasets* considerados, que en el caso de FCA son binarios mientras que en el modelo relacional son multivaluados. La noción de implicaciones en FCA puede ampliarse para poder tratar con contextos formales donde la relación I toma valores en un conjunto en lugar de ser binaria. Este tipo de contextos formales se denominan multivaluados y su aproximación se realiza a través del llamado análisis de conceptos formales difusos [8, 18], lo que queda fuera del ámbito de esta tesis doctoral.

2.3. Lógica de Implicaciones

En esta sección se van a introducir las lógicas de implicaciones que permitirán disponer de sistemas axiomáticos correctos y completos para trabajar con ellas. Para ello, el punto de partida van a ser los axiomas de Armstrong [4] como pionero en el campo. Sin embargo, el sistema de Armstrong no es adecuado para el razonamiento automático debido a su fuerte dependencia a la transitividad. En consecuencia, tras los Axiomas de Armstrong se presenta la lógica SL_{FD} , la cual sí va a permitir desarrollar métodos de razonamiento automático.

Ambas lógicas pueden aplicarse con éxito para trabajar con implicaciones y con DFs. El lenguaje sobre el que van a trabajar ambas lógicas es el mismo (ver Definición 2.1.8). Del mismo modo, la semántica puede verse en sendas definiciones, 2.1.9, 2.2.3, para implicaciones y DFs respectivamente. A continuación, se analizarán los sistemas axiomáticos y métodos de razonamiento. Para los siguientes apartados, y por facilitar la redacción y lectura, se utiliza el término implicaciones, siendo igualmente válido para DFs.

2.3.1. Axiomas de Armstrong

Los denominados axiomas de Armstrong son el primer sistema axiomático descrito para tratar sistemas de implicaciones utilizando la lógica [4]. Este sistema ha tenido una clara influencia en el diseño de varias lógicas sobre implicaciones, todas ellas construidas alrededor del paradigma de la transitividad [6, 58, 93].

Este sistema axiomático está formado por un esquema de axioma, denominado *Reflexivo*:

$$[\text{Ref}] \frac{}{AB \rightarrow A}$$

y dos reglas de inferencia, *Aumentativa* y *Transitiva* que se definen como:

$$[\text{Aug}] \frac{A \rightarrow B}{AC \rightarrow BC} \quad [\text{Tran}] \frac{A \rightarrow B, B \rightarrow C}{A \rightarrow C}$$

Para el sistema axiomático anterior, la noción de *deducción* (\vdash) se define como:

Definición 2.3.1. *Se dice que una implicación $A \rightarrow B$ se deriva sintácticamente (o se deduce) de un sistema de implicaciones Σ , y se denota por $\Sigma \vdash A \rightarrow B$, si existe una secuencia de implicaciones $\sigma_1, \dots, \sigma_n \in \mathcal{L}_M$ tal que $\sigma_n = A \rightarrow B$ y, para todo $1 \leq i \leq n$, la implicación σ_i satisface una de las siguientes condiciones:*

- σ_i es un axioma, es decir, verifica el esquema [Ref].
- $\sigma_i \in \Sigma$.
- σ_i se obtiene a partir de implicaciones pertenecientes a $\{\sigma_j \mid 1 \leq j < i\}$ aplicando las reglas de inferencia del sistema axiomático.

La secuencia $\sigma_1, \dots, \sigma_n$ constituye una demostración para $\Sigma \vdash A \rightarrow B$

Los axiomas de Armstrong son un sistema axiomático correcto y completo tanto si se está trabajando con implicaciones en FCA [4] como con DFs en bases de datos relacionales [46].

Debido al rol central que desempeña la transitividad en ese sistema axiomático, el desarrollo de métodos ejecutables para resolver problemas de implicaciones se ha mostrado infructuoso y se ha tenido que recurrir a métodos indirectos como se verá en la Sección 2.4. No obstante, como se ha mencionado, es un buen punto de partida para el desarrollo de nuevas lógicas para el tratamiento de las implicaciones, en concreto, para la lógica SL_{FD} , que se introduce a continuación.

2.3.2. Lógica de Simplificación

La lógica SL_{FD} no toma el paradigma de la transitividad como centro sino que se guía por la idea de simplificar el conjunto de implicaciones mediante la eliminación de atributos redundantes de manera eficiente [26]. Por consiguiente, la introducción de la lógica SL_{FD} abrió la puerta al desarrollo de métodos de razonamiento automatizados directamente basados en su novedoso sistema axiomático [27, 29]. La posibilidad de desarrollar estas

aplicaciones ha sido la motivación principal para el desarrollo de esta tesis, buscando sobre todo, mejorar la eficacia y la eficiencia.

Seguidamente, se va a comenzar definiendo su sistema axiomático que, como ya se ha comentado, es válido para ambas semánticas (FCA, DFs) y además es común debido a que utilizan el mismo lenguaje.

Sistema Axiomático

SL_{FD} se define como el par (\mathcal{L}_M, S_{FD}) donde S_{FD} tiene el siguiente esquema de axioma:

$$[\text{Ref}] \quad \overline{AB \rightarrow A}$$

junto con las siguientes reglas de inferencia, denominadas *fragmentación*, *composición* y *simplificación* respectivamente.

$$[\text{Frag}] \quad \frac{A \rightarrow BC}{A \rightarrow B} \quad [\text{Comp}] \quad \frac{A \rightarrow B, C \rightarrow D}{AC \rightarrow BD}$$

$$[\text{Simp}] \quad \text{Si } A \subseteq C, A \cap B = \emptyset, \frac{A \rightarrow B, C \rightarrow D}{A(C \setminus B) \rightarrow D}$$

SL_{FD} , al igual que los axiomas de Armstrong, constituye una lógica correcta y completa tanto para implicaciones como DFs, tal y como los autores presentaron en [26] y como demuestra el siguiente teorema.

Teorema 2.3.2. *Sea M un conjunto finito no vacío de atributos, $\Sigma \subseteq \mathcal{L}_M$ y $A \rightarrow B \in \mathcal{L}_M$. Entonces, $\Sigma \models A \rightarrow B$ si y sólo si $\Sigma \vdash A \rightarrow B$.*

Se destaca que el lenguaje SL_{FD} considera como fórmulas válidas aquellas en donde cualquiera de sus dos partes puede ser el conjunto vacío, es decir, de los tipos $A \rightarrow \emptyset$ y $\emptyset \rightarrow A$, tal y como los autores discutieron en [26].

La principal ventaja de la lógica SL_{FD} radica en que las reglas de inferencia pueden considerarse reglas de equivalencia. Como consecuencia, se ha podido utilizar como núcleo principal para el desarrollo de métodos automáticos para diversas aplicaciones (e.g. obtener claves minimales,

cálculos del cierre) como se verá más adelante. Un estudio más detallado al respecto, incluyendo teoremas y sus demostraciones, puede verse en [87].

Para finalizar, estos sistemas axiomáticos se pueden utilizar para el desarrollo de métodos de razonamiento automático con los que resolver el denominado problema de la implicación tal y como se muestra en el siguiente apartado.

2.4. Razonamiento Automático

Una vez introducido el lenguaje de la lógica, la semántica (en las dos acepciones que se han utilizado en esta tesis) y el sistema axiomático, se llega ahora a la última sección de estos preliminares. En ella se va a abordar la demostración automática a través del problema de la implicación y haciendo uso del cierre de atributos sobre el sistema axiomático.

Básicamente, el problema de la implicación es el siguiente: dado un conjunto de implicaciones Σ , determinar mediante la lógica si a partir de ese conjunto se puede inferir la validez de una nueva implicación dada $\Sigma \vdash A \rightarrow B$ [7, 76, 85].

A continuación se introduce el cierre de atributos, que se utiliza como base para construir el demostrador automático.

Definición 2.4.1. *Dado un conjunto $X \subseteq M$, llamamos cierre de X sobre Σ (notado X_{Σ}^+) como el mayor subconjunto de M tal que $\Sigma \vdash X \rightarrow X_{\Sigma}^+$.*

La definición anterior de cierre sintáctico de atributos da lugar a un operador de cierre, siempre y cuando el conjunto Σ sea un sistema de implicaciones completo (las implicaciones deducibles del contexto formal son todas inferibles desde Σ). No se entra en detalle sobre estos aspectos formales en estos preliminares puesto que, en todo el trabajo, se parte de la noción de cierre sintáctico como base.

En [87] los autores presentan un novedoso método para calcular el cierre de atributos sobre un conjunto de implicaciones, basado en la lógica SL_{FD} .

Se pasa ahora a presentar los métodos de razonamiento utilizados en esta tesis doctoral: el cierre clásico de Maier [75] como trabajo seminal

de este área, y a continuación, el cierre SL_{FD} , este último como núcleo principal del funcionamiento de los métodos diseñados y utilizados en las publicaciones que avalan este trabajo de tesis [13–15].

Para abordar el problema de la implicación, se utiliza el cierre sintáctico a modo de demostrador automático, pues dado un conjunto de implicaciones Σ , debido a la definición de cerrado anterior, se tiene que $\Sigma \vdash A \rightarrow B$ si y sólo si $B \subseteq A_{\Sigma}^{+}$. Este resultado hace que la definición de un método para calcular el cierre sintáctico abra la puerta al tratamiento automático del problema de la implicación. Ello será el centro del siguiente apartado.

2.4.1. Algoritmos para el Cálculo del Cierre

El problema de la implicación se ha abordado tradicionalmente utilizando un método básico que recibe como entrada un conjunto de atributos $X \subseteq M$ y un conjunto de implicaciones $\Sigma \subseteq \mathcal{L}_M$ y utiliza de forma exhaustiva la relación de subconjunto recorriendo iterativamente Σ y agregando nuevos elementos al cierre. Este método, fue propuesto en la década de 1970 por Maier [75] y puede considerarse la base principal donde se han sustentado tantos otros. Su funcionamiento puede verse en detalle en el Algoritmo 2.1.

Algoritmo 2.1: Cierre clásico

Entrada: Σ, A
Salida: A_{Σ}^{+}

```

1  inicio
2  |    $A_{\Sigma}^{+} := A$ 
3  |   repetir
4  |   |    $A' := A_{\Sigma}^{+}$ 
5  |   |   para cada  $X \rightarrow Y \in \Sigma$  hacer
6  |   |   |   si  $(X \subseteq A_{\Sigma}^{+})$  y  $(Y \not\subseteq A_{\Sigma}^{+})$  entonces
7  |   |   |   |    $A_{\Sigma}^{+} := A_{\Sigma}^{+} \cup \{Y\}$ 
8  |   |   hasta  $A_{\Sigma}^{+} = A'$ 
9  |   devolver  $A_{\Sigma}^{+}$ 

```

En [93], los autores muestran que la complejidad del problema del cierre es $\mathcal{O}(|A| |\Sigma|)$. En [87] se presentó el método del cierre basado en la lógica SL_{FD} . En dicho trabajo además se muestra que dicho método presenta un mejor rendimiento que los métodos anteriores. Como novedad, la salida del nuevo método de cierre no es sólo el conjunto cerrado sino que también produce un conjunto de implicaciones que puede ser interpretado como el conocimiento que resta en el sistema y que complementa a los atributos que están en el cerrado.

El método de razonamiento automático en SL_{FD} se basa en el Teorema de la deducción y un conjunto de equivalencias. Introduciremos ambas cosas antes de presentar el método.

Siguiendo la forma habitual, dos sistemas de implicaciones $\Sigma_1, \Sigma_2 \subseteq \mathcal{L}_M$ se dicen equivalentes si para toda implicación $A \rightarrow B$ de Σ_1 , se tiene que $\Sigma_2 \vdash A \rightarrow B$ y viceversa.

Teorema 2.4.2 (Teorema de la deducción). *Sea $A \rightarrow B \in \mathcal{L}_M$ y $\Sigma \subseteq \mathcal{L}_M$. Entonces,*

$$\Sigma \vdash A \rightarrow B \quad \text{si y sólo si} \quad \{\emptyset \rightarrow A\} \cup \Sigma \vdash \{\emptyset \rightarrow B\}$$

La siguiente proposición proporciona tres equivalencias, denominadas también: *Fragmentación, Composición y Simplificación*.

Proposición 2.4.3. *Sean $A, B, C, D \subseteq M$. Se verifican las siguientes equivalencias:*

- (I) $\{A \rightarrow B\} \equiv \{A \rightarrow B \setminus A\}$
- (II) $\{A \rightarrow B, A \rightarrow C\} \equiv \{A \rightarrow B \cup C\}$
- (III) $\{A \rightarrow B, C \rightarrow D\} \equiv \{A \rightarrow B, C \setminus B \rightarrow D \setminus B\}$ *siendo $A \cap B = \emptyset$ y $A \subseteq C$*

Mediante el Teorema de la deducción y las equivalencias anteriores, se puede pasar de las reglas de inferencia del sistema axiomático clásico de Armstrong a un sistema que puede ser automatizable. En este punto aparece el cierre SL_{FD} , que los autores denominaron **C1s** [87], y que actúa según el siguiente procedimiento.

Los pasos del algoritmo desglosados en lenguaje natural y de forma más detallada son:

- (I) En primer lugar, se incluye la fórmula $\emptyset \rightarrow A$ en Σ y se usa como semilla por el método de razonamiento mediante las equivalencias mencionadas en la proposición anterior.
- (II) A continuación, el algoritmo entra en un proceso iterativo en el cual se irán aplicando las siguientes equivalencias hasta alcanzar un punto en el que no sea posible aplicar ninguna.
 - **Eq. I:** Si $B \subseteq A$ entonces $\{\emptyset \rightarrow A, B \rightarrow C\} \equiv \{\emptyset \rightarrow A \cup C\}$.
 - **Eq. II:** Si $C \subseteq A$ entonces $\{\emptyset \rightarrow A, B \rightarrow C\} \equiv \{\emptyset \rightarrow A\}$.
 - **Eq. III:** En otro caso $\{\emptyset \rightarrow A, B \rightarrow C\} \equiv \{\emptyset \rightarrow A, B \setminus A \rightarrow C \setminus A\}$.
- (III) En el momento en el que no sea posible aplicar ninguna de las equivalencias anteriores, el algoritmo termina.

Formalmente, el algoritmo **C1s** puede verse detalladamente en el pseudocódigo 2.2.

Aunque es cierto que existen en la literatura numerosas propuestas de algoritmos para calcular el cierre (la mayoría de ellas como modificaciones del cierre clásico de Maier [75]), la principal novedad y ventaja que aporta **C1s** es que, aparte del cálculo del cierre, y de manera simultánea, también se calcula el conjunto reducido de implicaciones, guardando de esta forma el conocimiento complementario que describe la información que no pertenece al cierre. Este hecho conduce sin lugar a dudas a una posición privilegiada, ya que evita el elevado coste de minería de datos para extraer el nuevo conjunto de implicaciones para el *dataset* reducido después de cada aplicación del método, algo imprescindible con las implementaciones clásicas. Esta característica del algoritmo es la que se ha explotado ampliamente en los resultados obtenidos a lo largo de esta tesis doctoral.

Para finalizar este apartado, se muestra el siguiente ejemplo básico de aplicación del algoritmo **C1s** propuesto.

Algoritmo 2.2: Cierre Cls**Entrada:** Σ, A **Salida:** A_{Σ}^+, Σ'

```

1  inicio
2  |    $A_{\Sigma}^+ := A$ 
3  |    $\Sigma' := \Sigma$ 
4  |   repetir
5  |   |    $A' := A_{\Sigma}^+$ 
6  |   |   para cada  $B \rightarrow C \in \Sigma$  hacer
7  |   |   |   si  $B \subseteq A_{\Sigma}^+$  entonces
8  |   |   |   |   (Eq. I)
9  |   |   |   |    $A_{\Sigma}^+ := A_{\Sigma}^+ \cup \{B\}$ 
10 |   |   |   |    $\Sigma' := \Sigma' \setminus B \rightarrow C$ 
11 |   |   |   si no si  $C \subseteq A_{\Sigma}^+$  entonces
12 |   |   |   |   (Eq. II)
13 |   |   |   |    $\Sigma' := \Sigma' \setminus B \rightarrow C$ 
14 |   |   |   si no si  $(B \cap A_{\Sigma}^+ \neq \emptyset)$  o  $(C \cap A_{\Sigma}^+ \neq \emptyset)$  entonces
15 |   |   |   |   (Eq. III)
16 |   |   |   |    $\Sigma' := \Sigma' \cup \{B \setminus A_{\Sigma}^+ \rightarrow C \setminus A_{\Sigma}^+\}$ 
17 |   hasta  $A_{\Sigma}^+ = A'$ 
18 |   devolver  $A_{\Sigma}^+, \Sigma'$ 

```

Ejemplo 2.4.4. Sean $\Sigma = \{ak \rightarrow bc, cd \rightarrow gh, cij \rightarrow kl, de \rightarrow f, g \rightarrow de, hf \rightarrow ia, f \rightarrow c\}$ y $A = \{adf\}$.

La siguiente tabla muestra la traza de ejecución paso a paso del algoritmo.

	<i>Iteración 0</i>							
Σ'	$ak \rightarrow bc$	$cd \rightarrow gh$	$cij \rightarrow kl$	$de \rightarrow f$	$g \rightarrow de$	$hf \rightarrow ia$	$f \rightarrow c$	
A_{Σ}^+	adf							
	<i>Iteración 1</i>							
Σ'	$k \rightarrow bc$	$c \rightarrow gh$	$cij \rightarrow kl$	\times	$g \rightarrow e$	$h \rightarrow i$	\times	
Eq	III	III	-	II	III	III	I	
A_{Σ}^+	$acdf$							
	<i>Iteración 2</i>							
Σ'	$k \rightarrow b$	\times	$ij \rightarrow kl$		\times	\times		
Eq	III	I	III		I	I		
A_{Σ}^+		$acdfgh$			$acdefgh$		$acdefghi$	
	<i>Iteración 3</i>							
Σ'	$k \rightarrow b$		$j \rightarrow kl$					
Eq	-		-					
A_{Σ}^+	$acdefghi$							

El resultado del algoritmo **C1s** es:

- En primer lugar, el cierre $A_{\Sigma}^+ = \{a, c, d, e, f, g, h, i\}$.
- Y en segundo lugar y muy importante tal y como se ha mencionado, el nuevo conjunto reducido de implicaciones $\Sigma' = \{k \rightarrow b, j \rightarrow kl\}$, que contiene la información que no pertenece al cierre.



UNIVERSIDAD
DE MÁLAGA

Capítulo 3

Claves Minimales



UNIVERSIDAD
DE MÁLAGA

*Recuerde, mi amigo, que el conocimiento es más fuerte
que la memoria, y no debemos confiar en lo más débil.*

Drácula

B. Stoker

Título:	Reducing the search space by closure and simplification paradigms. A parallel key finding method
Autores:	Fernando Benito-Picazo, Pablo Cordero, Manuel Enciso, Ángel Mora
Revista:	The Journal of Supercomputing, Springer
Factor Impacto JCR:	1,349. Posición 52 de 104 (Q2)
Año:	2016
Categoría:	Computer Science, Theory & Methods
Publicación:	14 enero 2016
DOI:	10.1007/s11227-016-1622-1
Año:	2016



UNIVERSIDAD
DE MÁLAGA

Referencia completa

Benito-Picazo, F., Cordero, P., Enciso, M., and Mora, A. Reducing the search space by closure and simplification paradigms. *Journal of Supercomputing* vol. 73, 1 (Jan. 2017), pp. 75-87.

Resumen

In this paper, we present an innovative method to solve the minimal keys problem strongly based on the Simplification Logic for Functional Dependencies. This novel method improves previous logic-based methods by reducing, in a significant degree, the size of the search space this problem deals with. Furthermore, the new method has been designed to easily fit within a parallel implementation, thereby increasing the boundaries current methods can reach.

DOI

[10.1007/s11227-016-1622-1](https://doi.org/10.1007/s11227-016-1622-1)



UNIVERSIDAD
DE MÁLAGA

Capítulo 4

Generadores Minimales



UNIVERSIDAD
DE MÁLAGA

Delimitación..., ésta es una palabra a la que no temo, pues la labor de algo superior que posee el hombre, reside en un constante tender a limitar lo infinito, y en dividirlo y desintegrarlo en porciones perceptibles, es decir, en diferenciales.

Nosotros
Y. Zamiatin

Título:	Minimal generators, an affordable approach by means of massive computation
Autores:	Fernando Benito-Picazo, Pablo Cordero, Manuel Enciso, Ángel Mora
Revista:	The Journal of Supercomputing, Springer
Factor Impacto JCR:	1,532. Posición 44 de 103 (Q2)
Año:	2017
Categoría:	Computer Science, Theory & Methods
Publicación:	4 junio 2018
DOI:	10.1007/s11227-018-2453-z



UNIVERSIDAD
DE MÁLAGA

Referencia completa

Benito-Picazo, F., Cordero, P., Enciso, M., and Mora, A. Minimal generators, an affordable approach by means of massive computation. *The Journal of Supercomputing* (Online Jun 2018).

Resumen

Closed sets and minimal generators are fundamental elements to build a complete knowledge representation in formal concept analysis. The enumeration of all the closed sets and their minimal generators from a set of rules or implications constitutes a complex problem, drawing an exponential cost. Even for small datasets, such representation can demand an exhaustive management of the information stored as attribute implications. In this work, we tackle this problem by merging two strategies. On the one hand, we design a pruning, strongly based on logic properties, to drastically reduce the search space of the method. On the other hand, we consider a parallelization of the problem leading to a massive computation by means of a mapreduce like paradigm. In this study we have characterized the type of search space reductions suitable for parallelization. Also, we have analyzed different situations to provide an orientation of the resources (number of cores) needed for both the parallel architecture and the size of the problem in the splitting stage to take advantage in the map stage.

DOI

10.1007/s11227-018-2453-z



UNIVERSIDAD
DE MÁLAGA

Capítulo 5

Sistemas de Recomendación Conversacionales



UNIVERSIDAD
DE MÁLAGA

*Unas veces nacen los obstáculos de la
diversidad de las condiciones.
Sueño de una noche de verano
W. Shakespeare*

Título:	Enhancing the conversational process by using a logical closure operator in phenotypes implications
Autores:	Fernando Benito-Picazo, Manuel Enciso, Carlos Rossi, Antonio Guevara
Revista:	Mathematical Methods in the Applied Sciences, John Wiley & Sons Ltd.
Factor Impacto JCR:	1,180. Posición 91 de 252 (Q2)
Año:	2017
Categoría:	Mathematics, Applied
Publicación:	16 febrero 2017
DOI:	10.1002/mma.4338



UNIVERSIDAD
DE MÁLAGA

Referencia completa

Benito-Picazo, F., Enciso, M., Rossi, C., and Guevara, A. Enhancing the conversational process by using a logical closure operator in phenotypes implications. *Mathematical Methods in the Applied Sciences* vol. 41, 3 (2017), pp. 1089-1100.

Resumen

In this paper, we present a novel strategy to face the problem of dimensionality within datasets involved in conversational and feature selection systems. We base our work on a sound and complete logic along with an efficient attribute closure method to manage implications. All of them together allow us to reduce the overload of information we encounter when dealing with these kind of systems. An experiment carried out over a dataset containing real information comes to expose the benefits of our design.

DOI

10.1002/mma.4338



UNIVERSIDAD
DE MÁLAGA

Capítulo 6

Conclusiones y Trabajos Futuros



UNIVERSIDAD
DE MÁLAGA

*Si una conclusión no está poéticamente equilibrada,
no puede ser científicamente cierta.*

Los robots del amanecer

I. Asimov

Fundamentalmente, la naturaleza dual de esta tesis ha conllevado dos grandes grupos de tareas. Por un lado, se ha realizado un profundo estudio de los métodos basados en la lógica para el tratamiento eficiente de la información utilizando los conjuntos de implicaciones que se verifican en un determinado *dataset*. Y por otro lado, se han realizado una serie de tareas para contrastar la validez de estos métodos teóricos en la práctica.

Se ha investigado sobre tres problemas diferentes: claves minimales, generadores minimales y sistemas de recomendación. Para cada una de ellas se han realizado multitud de experimentos que demuestran la utilidad y la validez del trabajo realizado. Asimismo, se hace un especial hincapié en la parte aplicada del estudio con la intención de facilitar la transferencia de conocimiento a entornos diferentes del ámbito académico, como el mercado empresarial.

A lo largo de la tesis se puede apreciar el hecho de que contar con una sólida teoría basada en la Lógica y las Matemáticas concede la base

para la creación de métodos automatizados con los que poder afrontar el desarrollo de aplicaciones de ingeniería. De esta forma, se ha comprobado que existe una gran cantidad de información implícita en los datos que se suelen utilizar en dichas aplicaciones. El descubrimiento de toda esta información y su gestión inteligente es sin duda una clara oportunidad de investigación con una fuerte actividad y repercusión en la actualidad. Esta ha sido la intención principal a la hora de trabajar con FCA y los conjuntos de implicaciones. Pasar de la teoría a la práctica y viceversa ha sido uno de los principales desafíos de esta tesis al hacer que estos conceptos se conviertan en una herramienta fructífera para la representación, gestión y análisis del conocimiento en situaciones reales.

Se alcanza ahora el último capítulo de la tesis, en el cual se recopilan las conclusiones más importantes alcanzadas como resultado del trabajo de investigación realizado. Seguidamente, cerrarán el capítulo una serie de tareas con las que continuar a partir de este punto y que se presentan como trabajos futuros.

6.1. Conclusiones

En primer lugar, y dada la relación manifiesta en términos de algoritmos y computación paralela, se muestran las conclusiones referentes a claves y generadores minimales. Más adelante y para terminar la sección, se muestran aquellas conclusiones obtenidas en torno a SRs conversacionales.

Como se ha mencionado con anterioridad, conocer las claves es una tarea crucial para muchas áreas de gestión de la información. Se recuerda que una clave es un conjunto de atributos de un esquema relacional que nos permite distinguir inequívocamente cada objeto del *dataset*. El problema surge debido a la complejidad exponencial del algoritmo de búsqueda de claves a partir de un conjunto de DFs que se cumplen en un esquema del modelo relacional [74].

Para abordar este problema, a partir de los métodos que los autores presentaron en [25], se han diseñado una serie de nuevos métodos que permiten computar el conjunto de claves a partir del conjunto de implicaciones

haciendo uso de métodos de razonamiento automático basados en la lógica SL_{FD} . Se han diseñado e implementado, pasando desde la teoría a la práctica, los diferentes métodos basados en el paradigma de tableaux, y se ha verificado como, hasta donde se ha podido comprobar, los resultados obtenidos mejoran los de las aproximaciones anteriores [25] reduciendo tres aspectos fundamentales: el tiempo de cómputo, los cálculos redundantes y el tamaño del problema [13]. Además, en este trabajo se ha realizado el diseño y la implementación paralela del algoritmo original, poniendo de manifiesto como la introducción del paralelismo y los recursos de supercomputación permiten que las limitaciones que se encontraban en el trabajo original [25] hayan podido solventarse, abriendo la puerta a poder trabajar con cantidades mayores de información.

Por otro lado, enumerar todos los conjuntos cerrados y sus generadores minimales es también un problema muy complejo pero esencial en varias áreas de conocimiento, constituyendo una oportunidad para mostrar los beneficios de FCA cuando se trabaja en aplicaciones reales. Junto con los conjuntos cerrados, los generadores minimales, son esenciales para obtener una representación completa del conocimiento en FCA [96].

El punto de partida para trabajar sobre generadores minimales en esta tesis ha sido el método presentado en [28], donde se utilizó la lógica SL_{FD} como herramienta para encontrar todos los generadores minimales a partir de un conjunto de implicaciones. La propuesta que se ha realizado en esta tesis ha consistido en el diseño y la implementación de métodos que nos permitan identificar los generadores minimales como representaciones canónicas de cada conjunto cerrado para un conjunto de implicaciones.

Desafortunadamente, la dificultad que aparece al utilizar estos métodos es que la obtención de todos los conjuntos cerrados y sus respectivos generadores minimales es un problema con complejidad exponencial.

Con la intención de afrontar esta tarea, se han diseñado dos métodos de poda para mejorar el rendimiento de la enumeración de los generadores minimales. Para ello, se ha hecho un uso intensivo de la lógica SL_{FD} sobre conjuntos de implicaciones. Finalmente, se han diseñado, analizado y probado algoritmos diferentes (MinGen, MinGenPr, GenMinGen),

mostrando claramente las mejoras aportadas por cada uno. Así, se han alcanzado reducciones de más del 50 % en el número de nodos del árbol de búsqueda que construye el método original, MinGen, frente al resto, MinGenPr y GenMinGen, como se ha detallado en [14].

Asimismo, la utilización de estrategias paralelas se alza como la mejor alternativa en la resolución tanto de los problemas de claves minimales como de generadores minimales. Este hecho se debe a que cada uno de los subproblemas que se generan en la resolución de estos problemas (cada nodo del árbol del tableaux) es una instancia equivalente del problema original pero reducida, y por tanto, pueden tratarse de manera paralela asignando cada uno de ellos a un procesador diferente.

No obstante, el primer punto que es necesario aclarar es que se ha aplicado un *paralelismo* de tipo *hardware*. Es decir, ha consistido en la utilización de un conjunto de procesadores que se encargan de ir resolviendo cada uno de los subproblemas de forma simultánea. Por lo tanto, estas implementaciones no son un caso de desarrollo de código paralelo desde una visión más centrada en la programación, sino que es más acertado considerarlas como aplicaciones basadas en una estrategia *MapReduce* [35], que se ejecutan de forma paralela con la ayuda de recursos *hardware*.

Al igual que para el problema de las claves minimales, se ha desarrollado el código necesario para poder trabajar con grandes cantidades de información para identificar los generadores minimales. Para resolver los problemas de tiempo de ejecución cuando la cantidad de información de entrada sea considerable, el desarrollo se ha optimizado para ejecuciones que sean capaces de aprovechar grandes recursos computacionales, tales como los proporcionados por el Centro de Supercomputación y Bioinnovación de la Universidad de Málaga; gracias a ellos ha sido viable realizar la gran mayoría de las pruebas. Aun contando con dichos recursos, se ha llegado a la conclusión de que, para ambos casos, claves y generadores minimales, es absolutamente necesario que las implementaciones tengan en cuenta el correcto uso de los recursos de memoria; incluso para problemas pequeños, la cantidad de memoria que se puede necesitar puede dispararse sustancialmente.

Sin embargo, para la mayoría de los casos, existe un serio inconveniente. A la hora de afrontar la resolución de un problema de claves o generadores minimales, no es posible, en primera instancia, prever cuál va a ser la magnitud que va a alcanzar la resolución del problema (en términos de número de nodos y tiempo de ejecución) a la vista únicamente de la información de entrada. Esto obliga a realizar una serie de pruebas previas para configurar adecuadamente el entorno de ejecución en cuanto a número de procesadores, cantidad de memoria y espacio de almacenamiento (véase [10]). Eliminar la necesidad de dichas pruebas previas constituye sin duda un problema complejo de cara a futuras investigaciones, como se mencionará más adelante en el apartado de trabajos futuros.

Para evaluar las implementaciones realizadas (partiendo del hecho de que son implementaciones de algoritmos con el mismo orden de complejidad), no es suficiente con la comparación de los tiempos de ejecución de los métodos ya que este parámetro va a venir condicionado por la arquitectura *hardware* utilizada para llevar a cabo los experimentos. En este sentido, se han utilizado métricas adicionales: el número de nodos y el número de claves o generadores redundantes, que reflejan el tamaño del problema y la cantidad de cálculo superfluo realizado, respectivamente. Como muestran los experimentos presentados en [9–11, 13, 14], gracias a estas métricas se ha podido comprobar que, con el trabajo realizado en esta tesis, se ha conseguido en diferentes experimentos una reducción del número de nodos y/o del número de cálculos superfluos.

En cuanto a los SRs conversacionales, el trabajo realizado se ha enfocado en el uso de implicaciones y la lógica para subsanar determinados problemas que aparecen en este tipo de SR. En concreto, se ha abordado el denominado problema de la dimensionalidad que surge cuando se trabaja con una cantidad muy elevada de atributos, lo que dificulta la interacción del sistema con el usuario.

En este campo también son varias las conclusiones alcanzadas. A alto nivel, la más importante es que, efectivamente, el tratamiento de la información realizado por medio de implicaciones y la lógica SL_{FD} puede aplicarse con éxito al campo de los SRs. Este hecho ya era auspiciado por

la existencia de trabajos en la literatura de SRs que utilizan conceptos de FCA [70,71,109,139]; el trabajo en esta tesis refuerza esta línea, en concreto para los SRs conversacionales, proponiendo nuevos métodos con los que abordar problemas comunes de esos sistemas y mejorar las aproximaciones existentes.

Más específicamente, se ha aportado una novedosa aplicación del algoritmo del cierre 2.2 para afrontar el problema de la sobrecarga de la información. Así, un punto importante de esta aportación ha sido la de aprovechar el resultado dual que resulta del algoritmo del cierre 2.2, pues además del conjunto cerrado de atributos, el hecho de poder disponer del conjunto de implicaciones que recoge la semántica del sistema complementario a dicho conjunto, es aprovechado para volver a calcular de nuevo el conjunto de implicaciones, lo que tiene un coste exponencial.

La solución propuesta presenta un proceso conversacional de selección de elementos por parte del usuario a partir de los atributos de éstos. Este trabajo combina además características de sistemas basados en contenido con sistemas de recomendación basados en conocimiento mediante una gestión inteligente de las implicaciones teniendo como base el cierre 2.2.

Para concretar las conclusiones obtenidas, debe tenerse en cuenta que, tal y como se introdujo anteriormente en la Sección 1.3, existen numerosas opciones a la hora de evaluar el funcionamiento de un SR. Esto es razonable en tanto en cuanto el número de técnicas diferentes con las que trabajan los SRs es igualmente alto. Por tanto, es fundamental decidir qué métricas son oportunas de aplicar dependiendo del tipo de SR que se desee evaluar, pues evidentemente habrá casos en los que una métrica no tenga cabida para un tipo de SR determinado.

Dada la naturaleza del sistema desarrollado, las métricas que se han utilizado para evaluar el rendimiento están directamente relacionadas con el proceso de diálogo, como son el número de pasos de la conversación o el filtrado de atributos que se produce. Estas métricas devuelven resultados muy prometedores en ambos casos:

- (I) En cuanto al número de pasos del diálogo, se puede ver como en gran

parte de los experimentos realizados, la conversación necesita menos de 3 ó 4 pasos para alcanzar recomendaciones adecuadas aun cuando el número total de atributos sea considerable (más de 30 atributos). Estos buenos resultados quedan contrastados en [15], donde se han realizado pruebas con los conjuntos de implicaciones derivados de *datasets* con 100 atributos diferentes y en las que la mayoría de las conversaciones finalizan en 2 ó 3 pasos.

- (II) En cuanto al filtrado de atributos, los resultados son igualmente notables. Las pruebas mostradas en ese mismo artículo, demuestran cómo el uso del sistema conversacional ha evitado que el usuario tenga que interactuar con tan sólo entre el 5 % y el 20 % de atributos según el experimento (en el peor y mejor caso, respectivamente) reduciendo de esta forma la sobrecarga de información y mejorando la experiencia de usuario.

Estos resultados superan con creces los encontrados en la literatura con métricas de evaluación comparables, como [118] donde, incluso en pruebas con *datasets* con un menor número de atributos, se puede ver que por un lado, necesitan un número de pasos mayor, y por otro, que la reducción del número de atributos en la conversación es menor.

Para finalizar, se quiere destacar que las aplicaciones y métodos desarrollados para los tres campos son capaces de ir más allá del ámbito académico o de investigación. Las pruebas realizadas demuestran la viabilidad y la utilidad que las propuestas pueden aportar en entornos empresariales. Como muestras de ello, se pueden considerar como referencias los experimentos satisfactorios llevados a cabo sobre datos reales, como el caso de MovieLens y los repositorios de la UCI¹ para claves y generadores minimales, o el caso de la información real sobre enfermedades y fenotipos extraída de HPO² y OMIM³ en el caso de los sistemas de recomendación conversacional.

¹Universidad de California, Irvine (<https://archive.ics.uci.edu/ml/index.php>)

²Human Phenotype Ontology Consortium (<https://hpo.jax.org/app/>)

³Online Mendelian Inheritance in Man (<https://www.omim.org>)

6.2. Trabajos Futuros

Los resultados obtenidos motivan una serie de líneas de trabajo importantes para futuras investigaciones.

Respecto a las aportaciones referentes al tema de claves y generadores minimales existen varios aspectos con los que continuar a partir de este trabajo de investigación. Se irán introduciendo uno a uno sin perjuicio de que el orden de aparición denote una mayor o menos importancia.

En el momento actual no es posible prever cuál va a ser la magnitud que va a alcanzar la resolución del problema a la vista de la información de entrada, en términos de tiempo de cómputo y de recursos computacionales. Esto va a constituir en la mayoría de los casos un serio inconveniente ya que, para la realización de aplicaciones o pruebas, esta circunstancia no permite adelantar los requisitos de tiempo y recursos computacionales que serán necesarios. A pesar de ello, se ha observado que ciertas características, como el cardinal de la premisa o la conclusión en la implicación, suelen vaticinar patrones de comportamiento similares. En este sentido, el trabajo que se está llevando a cabo es investigar la motivación teórica de estos hechos empíricos, con la intención de poder identificar características que pronostiquen la complejidad que alcanzará una determinada ejecución del proceso.

Otro camino muy importante para continuar la investigación tanto para claves como para generadores minimales consiste en ahondar en la optimización del valor de corte o BOV. Recuérdese que el BOV es el valor a partir del cual la ejecución secuencial del código paralelo termina y se forman los diferentes subproblemas que serán resueltos en paralelo. El hecho de establecer un valor de corte adecuado es una tarea realmente compleja. En estos momentos se está estudiando la forma de expansión que tiene el tableaux con la intención de optimizar el valor de corte de manera que equilibre el trabajo realizado por cada procesador.

Otra línea de investigación sobre la que se está trabajando busca profundizar en el hecho de que aumentar el número de procesadores para la resolución de un problema no siempre redundan en una mejora del rendimien-

to. Hay casos en los que aumentar los recursos utilizados puede ser incluso contraproducente como se ha demostrado en las publicaciones que avalan esta tesis [13, 14]. En otras palabras, los tiempos de ejecución de los experimentos pueden incrementarse al aumentar los recursos *hardware*. Este efecto se debe, generalmente, a que el problema original se disemina de manera excesiva entre los procesadores disponibles provocando que el tiempo requerido por las comunicaciones para combinar los resultados parciales y así construir el resultado final contrarresta la ganancia en rendimiento que ofrece la capacidad de cómputo adicional. Para abordar este problema, se está trabajando en descubrir aquellas cotas de recursos a partir de las cuales el beneficio decrece. Para ello, se están investigando estrategias de optimización de recursos *hardware* en entornos de HPC [3, 126].

Otra tarea por la que continuar es realizar un nuevo diseño de los algoritmos paralelos que permita establecer comunicación entre las diferentes resoluciones paralelas de un mismo problema, de forma que se pueda mejorar la reducción de los cálculos redundantes. Para el caso concreto del cálculo de los generadores minimales, este objetivo es muy importante y la razón es la siguiente. Como se ha mencionado en el Capítulo 1, el método que mejores resultados obtiene en cuanto a número de nodos y tiempos de ejecución es GenMinGen, sin embargo, hay que recordar que, por el momento, es un método secuencial debido al requisito de establecer comunicación entre las diferentes soluciones parciales. Según el curso actual de la investigación, va a ser necesario considerar la aplicación del paralelismo a nivel de *software* y analizar cómo puede combinarse con la estrategia actual, centrada en el paralelismo *hardware*. Por tanto, se ha comenzado por investigar trabajos relacionados en la literatura sobre este tema [34].

En relación a los SRs conversacionales, aparecen varios aspectos interesantes a tener en cuenta en el futuro próximo.

En primer lugar, sería esencial averiguar qué características del *dataset* (tamaño, sobreespecialización, dispersión, sinónimos, etc.) son las más influyentes en el rendimiento del proceso de recomendación. En busca de este objetivo, en estos momentos se ha comenzado por investigar la literatura al respecto [19, 122].

Existen más aspectos de los SRs que sería recomendable investigar para mejorar el sistema conversacional desarrollado. Tal puede ser el caso de proporcionar explicaciones que justifiquen las recomendaciones que el usuario recibe. Este es un aspecto muy importante en un SR, ya que ayuda a mantener un mayor grado de confianza del usuario en los resultados generados por el sistema [111]. De hecho, la aceptación de un SR mejora cuando los usuarios comprenden las fortalezas y limitaciones del SR [92].

En este sentido, es relevante considerar que al tratar con implicaciones, nuestro SR garantiza que los resultados cumplen plenamente con lo que pide el usuario. Sin embargo, consideramos interesante la generación de explicaciones para justificar la reducción de atributos que se lleva a cabo a lo largo del proceso conversacional, al margen de los solicitados explícitamente por el usuario, por la acción del algoritmo del cierre.

Por todo ello, en este momento se está investigando a partir de la literatura los tres diferentes estilos de explicaciones habituales en los SRs actuales, que son:

- (I) Los basados en usuario y sus preferencias, donde las explicaciones se centran en justificar las recomendaciones realizadas argumentando afinidad entre usuarios y/o preferencias [132].
- (II) Los basados en objetos. En este caso, las explicaciones del SR se razonan teniendo en cuenta atributos añadidas de los objetos, como por ejemplo, su historial de valoraciones.
- (III) Los basados en características. Las explicaciones son similares al caso anterior, pero esta vez teniendo en cuenta las características intrínsecas de los objetos [86].

En el futuro próximo, el trabajo se centrará en el estudio del tercer tipo de explicaciones ya que, por el momento, los dos primeros tipos no tienen cabida en el sistema desarrollado, ya que no se utilizan ni preferencias de usuario ni valoraciones.

Índice alfabético

- SL_{FD} , 8–10, 13, 16, 19, 44, 45, 75
- Análisis Formal de Conceptos, 3
- Axiomas de Armstrong, 5, 43
- bases, 12
- bases de datos, 3
- Bases de Datos Relacionales, 39
- Cierre SL_{FD} , 48
- Cierre clásico, 47
- CK, 10
- clave, 6
 - claves minimales, 5, 8, 10, 13, 14, 20, 23, 29, 73
- concepto formal, 33
- conjunto de implicaciones, 3, 5, 10, 14, 15, 19, 23, 29, 46, 49, 73, 78
- contexto formal, 30–32, 34, 35, 38, 39, 42
- Cuerpo, 40
- dataset, 4, 5, 13, 18, 42, 81
- dependencias funcionales, 4, 41
- Esquema, 40
- generadores minimales, 5, 12, 14, 15, 23, 29, 73, 75, 81
- GenMinGen, 14
- implicaciones, 4, 8, 13, 16, 18, 20, 23, 29, 37–39, 42–44, 46, 77, 82
- Lógica de Simplificación, 5
- MapReduce, 11, 15, 76
- MinGen, 13
- MinGenPr, 13, 14
- Modelo, 38
- operador de cierre, 10, 33, 38
- operadores de derivación, 32
- paralelismo, 10, 11, 14, 75, 81
- problemas SR, 18
 - arranque en frío, 18
 - ataques maliciosos, 18
 - dimensionalidad, 18, 19
 - escalabilidad, 18
 - escasez, 18
 - oveja-negra, 18
 - postergación, 18



privacidad, 18
sobreespecialización, 18

reglas de inferencia, 13, 45
 composición, 45
 fragmentación, 45
 simplificación, 45

retículo de conceptos, 35, 36, 39
 conjuntos cerrados, 12

sistema axiomático, 43, 45

sistemas de recomendación, 3, 15,
 29, 73
 basados en conocimiento, 17
 basados en contenido, 17
 colaborativos, 17
 conversacionales, 17, 77

SST, 10

supercomputación, 5, 10, 12, 24,
 75

Tableaux, 9

Teorema de la deducción, 48

tupla, 6, 41

Índice de figuras

1. Producción científica	VII
1.1. Esquema del estado del arte y las contribuciones generadas.	22
1.2. Esquema de la estructura de la tesis y las publicaciones. . .	25
2.1. Esquema de contenido del Capítulo 2, Preliminares.	30
2.2. Retículo de conceptos asociado al contexto formal 2.1.2 . .	36



UNIVERSIDAD
DE MÁLAGA

Índice de tablas

2.1. Ejemplo de contexto formal sobre los destinos aéreos del grupo Star Alliance [45]	32
2.2. Extracto del ejemplo de contexto formal sobre los destinos aéreos del grupo Star Alliance 2.1	35



UNIVERSIDAD
DE MÁLAGA

Anexo



UNIVERSIDAD
DE MÁLAGA

Apéndice A

Closed sets enumeration: a logical approach

Referencia completa

Fernando Benito-Picazo, Pablo Cordero, Manuel Enciso, Ángel Mora. *Closed sets enumeration: a logical approach*. Proceedings of the Seventeenth International Conference on Computational and Mathematical Methods in Science and Engineering, CMMSE, 2017. Cádiz, Spain, July 4-8, pp. 287-292.

Resumen

Closed sets are the basis for the development of the concept lattice, a key issue in formal concept analysis. The enumeration of all the closed sets is a complex problem, having an exponential cost. In addition to the closed set, it is very useful for applications to add the information of all the minimal generators for each closed set. In this work we explain how to approach this problem from a complete set of implication by means of a sound and complete logic.

ISBN

ISBN: 978-84-617-8694-7



UNIVERSIDAD
DE MÁLAGA

Apéndice B

Conversational recommendation to avoid the cold-start problem

Referencia completa

Fernando Benito-Picazo, Manuel Enciso, Carlos Rossi, Antonio Guevara. *Conversational recommendation to avoid the cold-start problem*. Proceedings of the Sixteenth International Conference on Computational and Mathematical Methods in Science and Engineering, CMMSE, 2016. Cádiz, Spain, July 4-8, pp. 184-190.

Resumen

Recommender systems has become a widespread topic, allowing to connect user demands to those products more suitable to their preferences. The more information we provide to the system, the better the system works. This is a weak point of recommenders: they need an initial information belonging to each new user. In this paper we propose to avoid the so-called cold-start problem by using a conversational recommendation approach. We consider products characteristics as attributes and deal with the attribute implications by means of the simplification logic to guide the user in the search.

ISBN

ISBN: 978-84-608-6082-2



UNIVERSIDAD
DE MÁLAGA

Apéndice C

Keys for the fusion of heterogeneous information

Referencia completa

Fernando Benito-Picazo, Pablo Cordero, Manuel Enciso, Ángel Mora. *Keys for the fusion of heterogeneous information*. Proceedings of the Fifteenth International Conference on Computational and Mathematical Methods in Science and Engineering, CMMSE, 2015. Cádiz, Spain, July 6-10, pp. 201-211.

Resumen

The management of heterogeneous information is a current topic which demands the use of intelligent techniques to deal with data semantics. In this work we approach this problem by using Simplification Logic. It has a sound and complete inference system conceived to treat implications and functional dependencies. The automatic processing of functional dependencies allows to develop methods and tools to tackle most classical problems in database and information processing. In this work, we use Simplification Logic to design a method to enumerate all minimal keys of a data repository inferring them from a set of functional dependencies. We also illustrates how this method provides a successful way to solve some outstanding problems in data processing in linked data.

ISBN

ISBN: 978-84-617-2230-3



UNIVERSIDAD
DE MÁLAGA

Apéndice D

**Increasing the Efficiency of
Minimal Key Enumeration
Methods by Means of
Parallelism**

Referencia completa

Fernando Benito-Picazo, Pablo Cordero, Manuel Enciso, Ángel Mora. *Increasing the Efficiency of Minimal Key Enumeration Methods by Means of Parallelism*. Proceedings of the 9th International Conference on Software Engineering and Applications, ICSOFT-EA, Vienna, Austria, August 29-31, 2014, pp. 512-517.

Resumen

Finding all minimal keys in a table is a hard problem but also provides a lot of benefits in database design and optimization. Some of the methods proposed in the literature are based on logic and, more specifically on tableaux paradigm. The size of the problems such methods deal with is strongly limited, which implies that they cannot be applied to big database schemas. We have carried out an experimental analysis to compare the results obtained by these methods in order to estimate their limits. Although tableaux paradigm may be viewed as a search space guiding the key finding task, none of the previous algorithms have incorporated parallelism. In this work, we have developed two different versions of the algorithms, a sequential and a parallel one, stating clearly how parallelism could naturally be integrated and the benefits we get over efficiency. This work has also guided future work guidelines to improve future designs of these methods.

DOI

10.5220/0005108205120517



UNIVERSIDAD
DE MÁLAGA

Bibliografía

- [1] ADOMAVICIUS, G., AND TUZHILIN, A. Toward the next generation of recommender systems: A survey of the state-of-the-art and possible extensions. *IEEE Trans. on Knowl. and Data Eng.* 17, 6 (June 2005), 734–749.
- [2] ADOMAVICIUS, G., AND TUZHILIN, A. In *Recommender Systems Handbook*, F. Ricci, L. Rokach, B. Shapira, and P. B. Kantor, Eds. Springer, 2011, pp. 217–253.
- [3] AL-ALI, R., KATHIRESAN, N., ANBARI, M. E., SCHENDEL, E. R., AND ZAID, T. A. Workflow optimization of performance and quality of service for bioinformatics application in high performance computing. *Journal of Computational Science* 15 (2016), 3 – 10. International Computational Science and Engineering Conference 2015 (ICSEC15).
- [4] ARMSTRONG, W. W. Dependency structures of data base relationships. In *Proceedings of the International Federation for Information Processing Congress* (1974), pp. 580–583.
- [5] ARMSTRONG, W. W., AND DEOBEL, C. Decompositions and functional dependencies in relations. *ACM Trans. Database Syst.* 5, 4 (Dec. 1980), 404–430.



- [6] ATZENI, P., AND DE ANTONELLIS, V. *Relational Database Theory*. Benjamin-Cummings Publishing Co., Inc., Redwood City, CA, USA, 1993.
- [7] BEERI, C., AND VARDI, M. Y. The implication problem for data dependencies. In *Automata, Languages and Programming* (Berlin, Heidelberg, 1981), S. Even and O. Kariv, Eds., Springer Berlin Heidelberg, pp. 73–85.
- [8] BELOHLAVEK, R. Similarity relations in concept lattices. *J. Log. Comput.* 10, 6 (2000), 823–845.
- [9] BENITO-PICAZO, F. *Minimal Key-Par. Una versión paralela de los algoritmos de búsqueda de claves minimales basados en Tableaux*. Proyecto Fin de Carrera. Dpto. Lenguajes y Ciencias de la Computación, Universidad de Málaga, 2013.
- [10] BENITO-PICAZO, F. *Study of minimal keys algorithms based on tableaux methods. Increasing the range of treated problems by means of parallelism and prune strategies based on minimal subsets*. Master Thesis. Languages and Computer Science Department, Universidad de Málaga, 2014.
- [11] BENITO-PICAZO, F., CORDERO, P., ENCISO, M., AND MORA, A. Increasing the efficiency of minimal key enumeration methods by means of parallelism. In *ICSOFTEA 2014 - Proceedings of the 9th International Conference on Software Engineering and Applications, Vienna, Austria, 29-31 August, 2014* (2014), pp. 512–517.
- [12] BENITO-PICAZO, F., CORDERO, P., ENCISO, M., AND MORA, A. Closed sets enumeration: a logical approach. In *Proceedings of the Seventeenth International Conference on Computational and Mathematical Methods in Science and Engineering* (Cádiz, Spain, 2017), pp. 287–292.



- [13] BENITO-PICAZO, F., CORDERO, P., ENCISO, M., AND MORA, A. Reducing the search space by closure and simplification paradigms. *Journal of Supercomputing* 73, 1 (Jan. 2017), 75–87.
- [14] BENITO-PICAZO, F., CORDERO, P., ENCISO, M., AND MORA, A. Minimal generators, an affordable approach by means of massive computation. *The Journal of Supercomputing* (Online Jun 2018).
- [15] BENITO-PICAZO, F., ENCISO, M., ROSSI, C., AND GUEVARA, A. Enhancing the conversational process by using a logical closure operator in phenotypes implications. *Mathematical Methods in the Applied Sciences* 41, 3 (2017), 1089–1100.
- [16] BERTET, K., DEMKO, C., VIAUD, J.-F., AND GUÉRIN, C. Lattices, closures systems and implication bases: A survey of structural aspects and algorithms. *Theoretical Computer Science* 743 (2018), 93 – 109.
- [17] BOBADILLA, J., ORTEGA, F., HERNANDO, A., AND GUTIÉRREZ, A. Recommender systems survey. *Knowledge-Based Systems* 46 (2013), 109 – 132.
- [18] BURUSCO, A., AND FUENTES-GONZÁLEZ, R. Construction of the l-fuzzy concept lattice. *Fuzzy Sets and Systems* 97, 1 (1998), 109–114.
- [19] CANO, E., AND MORISIO, M. Characterization of public datasets for recommender systems. In *2015 IEEE 1st International Forum on Research and Technologies for Society and Industry Leveraging a better tomorrow (RTSI)* (Sept 2015), pp. 249–257.
- [20] CHEN, L., AND PU, P. Hybrid Critiquing-based Recommender Systems. In *Proceedings of the 12th International Conference on Intelligent User Interfaces* (New York, NY, USA, 2007), IUI '07, ACM, pp. 22–31.
- [21] CODD, E. F. A relational model of data for large shared data banks. *Commun. ACM* 13, 6 (1970), 377–387.



- [22] CODD, E. F. Further normalization of the data base relational model. *IBM Research Report, San Jose, California RJ909* (1971).
- [23] CODD, E. F. Normalized data base structure: A brief tutorial. In *Proceedings of the 1971 ACM SIGFIDET (Now SIGMOD) Workshop on Data Description, Access and Control* (New York, NY, USA, 1971), SIGFIDET '71, ACM, pp. 1–17.
- [24] CODD, E. F. *The Relational Model for Database Management: Version 2*. Addison-Wesley Longman Publishing Co., Inc., Boston, MA, USA, 1990.
- [25] CORDERO, P., ENCISO, M., AND MORA, A. Automated reasoning to infer all minimal keys. In *Proceedings of the Twenty-Third International Joint Conference on Artificial Intelligence* (2013), IJCAI'13, AAAI Press, pp. 817–823.
- [26] CORDERO, P., ENCISO, M., MORA, A., AND DE GUZMÁN, I. P. SLFD logic: Elimination of data redundancy in knowledge representation. In *IBERAMIA 2002: Proceedings of the 8th Ibero-American Conference on AI* (London, UK, 2002), Springer-Verlag, pp. 141–150.
- [27] CORDERO, P., ENCISO, M., MORA, A., AND DE GUZMÁN, I. P. A tableaux-like method to infer all minimal keys. *Logic Journal of the IGPL* 22, 6 (2014), 1019–1044.
- [28] CORDERO, P., ENCISO, M., MORA, A., AND OJEDA-ACIEGO, M. Computing minimal generators from implications: a logic-guided approach. In *Proceedings of The Ninth International Conference on Concept Lattices and Their Applications, Fuengirola (Málaga), Spain, October 11-14, 2012* (2012), pp. 187–198.
- [29] CORDERO, P., ENCISO, M., MORA, A., OJEDA-ACIEGO, M., AND ROSSI, C. Knowledge discovery in social networks by using a logic-based treatment of implications. *Knowledge-Based System* 87 (2015), 16–25.



- [30] COSTANTE, E., DEN HARTOG, J., PETKOVIC, M., ETALLE, S., AND PECHENIZKIY, M. A white-box anomaly-based framework for database leakage detection. *Journal of Information Security and Applications* 32 (2017), 27 – 46.
- [31] CRESPO, R. G., MARTÍNEZ, O. S., LOVELLE, J. M. C., GARCÍA-BUSTELO, B. C. P., GAYO, J. E. L., AND DE PABLOS, P. O. Recommendation system based on user interaction data applied to intelligent electronic books. *Computers in Human Behavior* 27, 4 (2011), 1445 – 1449. Social and Humanistic Computing for the Knowledge Society.
- [32] DARWEN, H., DATE, C. J., AND FAGIN, R. A normal form for preventing redundant tuples in relational databases. In *Proceedings of the 15th International Conference on Database Theory* (New York, NY, USA, 2012), ICDT '12, ACM, pp. 114–126.
- [33] DE CAMPOS, L. M., FERNÁNDEZ-LUNA, J. M., HUETE, J. F., AND RUEDA-MORALES, M. A. Combining content-based and collaborative recommendations: A hybrid approach based on bayesian networks. *International Journal of Approximate Reasoning* 51, 7 (2010), 785 – 799.
- [34] DE MORAES, N. R. M., DIAS, S. M., FREITAS, H. C., AND ZÁRATE, L. E. Parallelization of the next closure algorithm for generating the minimum set of implication rules. *Artificial Intelligence Research* 5 (03 2016), 40–54.
- [35] DEAN, J., AND GHEMAWAT, S. Mapreduce: Simplified data processing on large clusters. *Commun. ACM* 51, 1 (Jan. 2008), 107–113.
- [36] DU BOUCHER-RYAN, P., AND BRIDGE, D. Collaborative recommending using formal concept analysis. *Knowledge-Based Systems* 19, 5 (2006), 309 – 315.



- [37] EIRINAKI, M., GAO, J., VARLAMIS, I., AND TSERPES, K. Recommender systems for large-scale social networks: A review of challenges and solutions. *Future Generation Computer Systems* 78 (2018), 413–418.
- [38] ELMASRI, R., AND NAVATHE, S. *Fundamentals of Database Systems*, 6 ed. Prentice Hall International, 2010.
- [39] ENDRES, D., ADAM, R., GIESE, M. A., AND NOPPENNEY, U. Understanding the semantic structure of human fmri brain recordings with formal concept analysis. In *Proceedings of the 10th International Conference on Formal Concept Analysis* (Berlin, Heidelberg, 2012), ICFCA'12, Springer-Verlag, pp. 96–111.
- [40] FADOUS, R., AND FORSYTH, J. Finding candidate keys for relational data bases. In *SIGMOD Conference* (1975), W. F. King, Ed., ACM, pp. 203–210.
- [41] FAGIN, R. A normal form for relational databases that is based on domains and keys. *ACM Trans. Database Syst.* 6, 3 (Sept. 1981), 387–415.
- [42] FEIL, S., KRETZER, M., WERDER, K., AND MAEDCHE, A. Using gamification to tackle the cold-start problem in recommender systems. In *Proceedings of the 19th ACM Conference on Computer Supported Cooperative Work and Social Computing Companion* (New York, NY, USA, 2016), CSCW '16 Companion, ACM, pp. 253–256.
- [43] FENG, H., TIAN, J., WANG, H. J., AND LI, M. Personalized recommendations based on time-weighted overlapping community detection. *Information & Management* 52, 7 (2015), 789–800.
- [44] FRIEDMAN, A., KNIJNENBURG, B. P., VANHECKE, K., MARTENS, L., AND BERKOVSKY, S. *Privacy Aspects of Recommender Systems*. Springer US, Boston, MA, 2015, pp. 649–688.



- [45] GANTER, B., STUMME, G., AND WILLE, R. *Formal concept analysis: Methods and applications in computer science*. TU Dresden, 2002, pp. 1–85.
- [46] GANTER, B., AND WILLE, R. *Formal Concept Analysis: Mathematical Foundations*, 1st ed. Springer-Verlag New York, Inc., Secaucus, NJ, USA, 1997.
- [47] GAVALAS, D., KONSTANTOPOULOS, C., MASTAKAS, K., AND PANTZIOU, G. Mobile recommender systems in tourism. *Journal of Network and Computer Applications* 39 (2014), 319 – 333.
- [48] GOH, T. N. An information management paradigm for statistical experiments. *Quality and Reliability Eng. Int.* 26, 5 (2010), 487–494.
- [49] GRAS, B., BRUN, A., AND BOYER, A. Identifying grey sheep users in collaborative filtering: A distribution-based technique. In *Proceedings of the 2016 Conference on User Modeling Adaptation and Personalization* (New York, NY, USA, 2016), UMAP '16, ACM, pp. 17–26.
- [50] GRIOL, D., AND MOLINA, J. M. Building multi-domain conversational systems from single domain resources. *Neurocomputing* 271 (2018), 59 – 69.
- [51] GUNAWARDANA, A., AND SHANI, G. A survey of accuracy evaluation metrics of recommendation tasks. *J. Mach. Learn. Res.* 10 (Dec. 2009), 2935–2962.
- [52] GUNAWARDANA, A., AND SHANI, G. *Evaluating Recommender Systems*. Springer US, Boston, MA, 2015, pp. 265–308.
- [53] GUO, G. Resolving data sparsity and cold start in recommender systems. In *Proceedings of the 20th International Conference on User Modeling, Adaptation, and Personalization* (Berlin, Heidelberg, 2012), UMAP'12, Springer-Verlag, pp. 361–364.



- [54] HARPER, F. M., AND KONSTAN, J. A. The movielens datasets: History and context. *ACM Trans. Interact. Intell. Syst.* 5, 4 (Jan. 2016), 19:1–19:19.
- [55] HERLOCKER, J. L., KONSTAN, J. A., TERVEEN, L. G., AND RIEDL, J. T. Evaluating collaborative filtering recommender systems. *ACM Trans. Inf. Syst.* 22, 1 (Jan. 2004), 5–53.
- [56] HILL, W., STEAD, L., ROSENSTEIN, M., AND FURNAS, G. Recommending and evaluating choices in a virtual community of use. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems* (New York, NY, USA, 1995), CHI '95, ACM Press/Addison-Wesley Publishing Co., pp. 194–201.
- [57] HUHTALA, Y., KARKKAINEN, J., PORKKA, P., AND TOIVONEN, H. Tane: An efficient algorithm for discovering functional and approximate dependencies. *Comput. J.* 42, 2 (1999), 100–111.
- [58] IBARAKI, T., KOGAN, A., AND MAKINO, K. Functional dependencies in horn theories. *Artificial Intelligence* 108, 1 (1999), 1 – 30.
- [59] ISINKAYE, F., FOLAJIMI, Y., AND OJOKOH, B. Recommendation systems: Principles, methods and evaluation. *Egyptian Informatics Journal* 16, 3 (2015), 261 – 273.
- [60] JANNACH, D., ZANKER, M., AND FUCHS, M. Constraint-Based Recommendation in Tourism: A Multiperspective Case Study. *Information Technology & Tourism* 11, 2 (2009), 139–155.
- [61] KAMRA, A., TERZI, E., AND BERTINO, E. Detecting anomalous access patterns in relational databases. *The VLDB Journal* 17, 5 (Aug. 2008), 1063–1077.
- [62] KARNAUGH, M. The map method for synthesis of combinational logic circuits. *American Institute of Electrical Engineers, Part I: Communication and Electronics, Transactions of the* 72, 5 (Nov. 1953), 593–599.



- [63] KEMPER, A., AND MOERKOTTE, G. Query optimization in object bases: Exploiting relational techniques. In *Query Processing for Advanced Database Systems, Dagstuhl*. Morgan Kaufmann, 1991, pp. 63–98.
- [64] KENT, W. A simple guide to five normal forms in relational database theory. *Commun. ACM* 26, 2 (Feb. 1983), 120–125.
- [65] KESTER, Q.-A. Application of formal concept analysis to visualization of the evaluation of risks matrix in software engineering projects. *International Journal of Science, Engineering and Technology Research (IJSETR)* 2 (Jan 2013), 220–225.
- [66] KIM, K. An improved semi-supervised dimensionality reduction using feature weighting: Application to sentiment analysis. *Expert Systems with Applications* 109 (2018), 49 – 65.
- [67] KUZNETSOV, S., AND OBIEDKOV, S. Comparing performance of algorithms for generating concept lattices. *Journal of Experimental and Theoretical Artificial Intelligence* 14 (2002), 189–216.
- [68] LAMPROPOULOS, A. S., LAMPROPOULOU, P. S., AND TSIHRINTZIS, G. A. A cascade-hybrid music recommender system for mobile services based on musical genre classification and personality diagnosis. *Multimedia Tools Appl.* 59, 1 (2012), 241–258.
- [69] LEE, S., AND CHOI, J. Enhancing user experience with conversational agent for movie recommendation: Effects of self-disclosure and reciprocity. *International Journal of Human-Computer Studies* 103 (2017), 95 – 105.
- [70] LEIVA, J. L., ENCISO, M., ROSSI, C., CORDERO, P., MORA, A., AND GUEVARA, A. Context-aware recommendation using fuzzy formal concept analysis. In *ICSOFIT* (2013), J. Cordeiro, D. A. Marca, and M. van Sinderen, Eds., SciTePress, pp. 617–623.



- [71] LEIVA, J. L., ENCISO, M., ROSSI, C., CORDERO, P., MORA, A., AND GUEVARA, A. Improving recommender systems with simplification logic to manage implications with grades. In *Software Technologies - 8th International Joint Conference, ICSOFT 2013, Reykjavik, Iceland, July 29-31, 2013, Revised Selected Papers (2013)*, pp. 290–305.
- [72] LINDEN, G., SMITH, B., AND YORK, J. Amazon.com recommendations: Item-to-item collaborative filtering. *IEEE Internet Computing* 7 (2003), 76–80.
- [73] LOPS, P., DE GEMMIS, M., AND SEMERARO, G. Content-based recommender systems: State of the art and trends. In *Recommender Systems Handbook*, F. Ricci, L. Rokach, B. Shapira, and P. B. Kantor, Eds. Springer, 2011, pp. 73–105.
- [74] LUCCHESI, C. L., AND OSBORN, S. L. Candidate keys for relations. *J. Comput. Syst. Sci.* 17, 2 (1978), 270–279.
- [75] MAIER, D. *The Theory of Relational Databases*. Computer Science Press, Rockville, 1983.
- [76] MAIER, D., MENDELZON, A. O., AND SAGIV, Y. Testing implications of data dependencies. *ACM Trans. Database Syst.* 4, 4 (Dec. 1979), 455–469.
- [77] MANDL, M., FELFERNIG, A., TEPPAN, E., AND SCHUBERT, M. Consumer decision making in knowledge-based recommendation. *Journal of Intelligent Information Systems* 37 (2011), 1–22.
- [78] MANOLOPOULOS, Y., THEODORIDIS, Y., AND TSOTRAS, V. J. *Advanced Database Indexing*, vol. 17 of *Advances in Database Systems*. Kluwer, 1999.
- [79] MCEANEANEY, W. M., DESHPANDE, A., AND GAUBERT, S. Curse-of-complexity attenuation in the curse-of-dimensionality-free method



for hjb pdes. In *2008 American Control Conference* (June 2008), pp. 4684–4690.

- [80] MCSHERRY, D. Minimizing dialog length in interactive case-based reasoning. In *Procs of the 17th Int Joint Conf on AI, IJCAI, Seattle, Washington, USA, August 4-10, 2001* (2001), pp. 993–998.
- [81] MEDINA-MOREIRA, J., APOLINARIO, O., LUNA-AVEIGA, H., LAGOS-ORTIZ, K., PAREDES-VALVERDE, M. A., AND VALENCIA-GARCÍA, R. A collaborative filtering based recommender system for disease self-management. In *Technologies and Innovation* (Cham, 2017), R. Valencia-García, K. Lagos-Ortiz, G. Alcaraz-Mármol, J. Del Cioppo, N. Vera-Lucio, and M. Bucaram-Leverone, Eds., Springer International Publishing, pp. 60–71.
- [82] MIMOUNI, N., NAZARENKO, A., AND SALOTTI, S. *A Conceptual Approach for Relational IR: Application to Legal Collections*. Springer International Publishing, Cham, 2015, pp. 303–318.
- [83] MISSAOUI, R., NOURINE, L., AND RENAUD, Y. An inference system for exhaustive generation of mixed and purely negative implications from purely positive ones. In *Proceedings of the 7th International Conference on Concept Lattices and Their Applications, Sevilla, Spain, October 19-21, 2010* (2010), pp. 271–282.
- [84] MISSAOUI, R., NOURINE, L., AND RENAUD, Y. Computing implications with negation from a formal context. *Fundam. Inf.* 115, 4 (Dec. 2012), 357–375.
- [85] MITCHELL, J. C. The implication problem for functional and inclusion dependencies. *Information and Control* 56, 3 (1983), 154 – 173.
- [86] MOONEY, R. J., AND ROY, L. Content-based book recommending using learning for text categorization. In *Proceedings of the Fifth*

ACM Conference on Digital Libraries (New York, NY, USA, 2000), DL '00, ACM, pp. 195–204.

- [87] MORA, A., CORDERO, P., ENCISO, M., FORTES, I., AND AGUILERA, G. Closure via functional dependence simplification. *International Journal of Computer Mathematics* 89, 4 (2012), 510–526.
- [88] MORGAN, C. G. An automated theorem prover for relational logic (abstract). In *Workshop Theorem Proving with Analytic Tableaux and Related Methods, Lautenbach. Universität Karlsruhe, Fakultät für Informatik, Institut für Logik, Komplexität und Deduktionssysteme, Interner Bericht 8/92, March 18-20, 1992* (1992), pp. 56–58.
- [89] NAGLER, T., AND CZADO, C. Evading the curse of dimensionality in nonparametric density estimation with simplified vine copulas. *Journal of Multivariate Analysis* 151 (2016), 69 – 89.
- [90] NIKOLOV, A., FERRARA, A., AND SCHARFFE, F. Data linking for the semantic web. *Int. J. Semant. Web Inf. Syst.* 7, 3 (July 2011), 46–76.
- [91] OUTRATA, J., AND VYCHODIL, V. Fast algorithm for computing fixpoints of galois connections induced by object-attribute relational data. *Information Sciences* 185, 1 (2012), 114 – 127.
- [92] PAPADIMITRIOU, A., SYMEONIDIS, P., AND MANOLOPOULOS, Y. A generalized taxonomy of explanations styles for traditional and social recommender systems. *Data Min. Knowl. Discov.* 24, 3 (2012), 555–583.
- [93] PAREDAENS, J., BRA, P., GYSSENS, M., AND GUCHT, D. V., Eds. *The structure of the relational database model*. EATCS Monographs on Theoretical Computer Science, 1989.
- [94] PERNELLE, N., SAÏS, F., AND SYMEONIDOU, D. An automatic key discovery approach for data linking. *Web Semantics: Science, Services and Agents on the WWW* 23 (2013), 16–30.



- [95] POELMANS, J., DEDENE, G., SNOECK, M., AND VIAENE, S. An iterative requirements engineering framework based on formal concept analysis and c k theory. *Expert Systems with Applications* 39, 9 (2012), 8115 – 8135.
- [96] POELMANS, J., IGNATOV, D. I., KUZNETSOV, S. O., AND DEDENE, G. Formal concept analysis in knowledge processing: A survey on applications. *Expert Systems with Applications* 40, 16 (2013), 6538 – 6560.
- [97] PORCEL, C., TEJEDA-LORENTE, A., MARTÍNEZ, M. A., AND HERRERA-VIEDMA, E. A hybrid recommender system for the selective dissemination of research resources in a technology transfer office. *Inf. Sci.* 184, 1 (Feb. 2012), 1–19.
- [98] QU, K., ZHAI, Y., LIANG, J., AND CHEN, M. Study of decision implications based on formal concept analysis. *International Journal of General Systems* 36, 2 (2007), 147–156.
- [99] REDA, A., PARK, Y., TIWARI, M., POSSE, C., AND SHAH, S. Metaphor: a system for related search recommendations. In *CIKM* (2012), X. wen Chen, G. Lebanon, H. Wang, and M. J. Zaki, Eds., ACM, pp. 664–673.
- [100] RESNICK, P., AND VARIAN, H. R. Recommender systems. *Commun. ACM* 40, 3 (Mar. 1997), 56–58.
- [101] REUSENS, M., LEMAHIEU, W., BAESENS, B., AND SELS, L. Evaluating recommendation and search in the labor market. *Knowledge-Based Systems* 152 (2018), 62 – 69.
- [102] RICCI, F., ROKACH, L., AND SHAPIRA, B. *Recommender Systems Handbook*, 2nd ed. Springer Publishing Company, 2015.
- [103] RISCH, V., AND SCHWIND, C. Tableaux-based theorem proving and non-standard reasoning. In *Workshop Theorem Proving with Analytic Tableaux and Related Methods, Lautenbach. Universität Karlsruhe*,



Fakultät für Informatik, Institut für Logik, Komplexität und Deduktionssysteme, Interner Bericht 8/92, March 18-20, 1992 (1992), pp. 76–78.

- [104] SACAREA, C., SOTROPA, D., AND TROANCA, D. Symptoms investigation by means of formal concept analysis for enhancing medical diagnoses. In *2017 25th International Conference on Software, Telecommunications and Computer Networks (SoftCOM)* (Sept 2017), pp. 1–5.
- [105] SAIEDIAN, H., AND SPENCER, T. An efficient algorithm to compute the candidate keys of a relational database schema. *Comput. J.* *39*, 2 (1996), 124–132.
- [106] SALI, A. *Minimal Keys in Higher-Order Datamodels*. Springer Berlin Heidelberg, Berlin, Heidelberg, 2004, pp. 242–251.
- [107] SALIMI, A., ZIAII, M., AMIRI, A., ZADEH, M. H., KARIMPOULI, S., AND MORADKHANI, M. Using a feature subset selection method and support vector machine to address curse of dimensionality and redundancy in hyperion hyperspectral data classification. *The Egyptian Journal of Remote Sensing and Space Science* *21*, 1 (2018), 27–36.
- [108] SALLAM, A., BERTINO, E., HUSSAIN, S. R., LANDERS, D., LEFLER, R. M., AND STEINER, D. Dbsafe: An anomaly detection system to protect databases from exfiltration attempts. *IEEE Systems Journal* *11*, 2 (June 2017), 483–493.
- [109] SENATORE, S., AND PASI, G. Lattice navigation for collaborative filtering by means of (fuzzy) formal concept analysis. In *Proceedings of the 28th Annual ACM Symposium on Applied Computing* (New York, NY, USA, 2013), SAC '13, ACM, pp. 920–926.
- [110] SHAH KHUSRO, ZAFAR ALI, I. U. Recommender systems: Issues, challenges, and research opportunities. In *Information Science and Applications (ICISA) 2016* (2016), vol. 376, IEEE, pp. 1179–1189.

- [111] SHARMA, R., AND RAY, S. Explanations in recommender systems: An overview. *Int. J. Bus. Inf. Syst.* 23, 2 (Jan. 2016), 248–262.
- [112] SIMSION, G. C., AND WITT, G. C. *Data modeling essentials*, 3rd ed. Amsterdam; Boston, 2005.
- [113] SISMANIS, Y., BROWN, P., HAAS, P. J., AND REINWALD, B. Gordian: efficient and scalable discovery of composite keys. In *In Proc. International Conference on Very Large Data Bases (VLDB (2006))*, pp. 691–702.
- [114] SON, J., AND KIM, S. B. Academic paper recommender system using multilevel simultaneous citation networks. *Decision Support Systems* 105 (2018), 24 – 33.
- [115] SON, L. H. Dealing with the new user cold-start problem in recommender systems: A comparative review. *Information Systems* 58 (2016), 87 – 104.
- [116] SUNDARESAN, N. Recommender systems at the long tail. In *Proceedings of the Fifth ACM Conference on Recommender Systems (New York, NY, USA, 2011)*, RecSys '11, ACM, pp. 1–6.
- [117] TIROSHI, A., KUFLIK, T., KAY, J., AND KUMMERFELD, B. Recommender systems and the social web. In *UMAP Workshops (2011)*, L. Ardissono and T. Kuffik, Eds., vol. 7138 of *Lecture Notes in Computer Science*, Springer, pp. 60–70.
- [118] TRABELSI, W., WILSON, N., BRIDGE, D. G., AND RICCI, F. Preference dominance reasoning for conversational recommender systems: a comparison between a comparative preferences and a sum of weights approach. *International Journal on Artificial Intelligence Tools* 20, 4 (2011), 591–616.
- [119] TU, X., WANG, Y., ZHANG, M., AND WU, J. Using formal concept analysis to identify negative correlations in gene expression data.



IEEE-ACM Transactions on Computational Biology and Bioinformatics 13 (Jun 2015), 1 – 1.

- [120] ULLMAN, J. D., AND WIDOM, J. *A First Course in Database Systems*. Prentice-Hall, Inc., Upper Saddle River, NJ, USA, 1997.
- [121] VAVILIS, S., EGNER, A., PETKOVIC, M., AND ZANNONE, N. An anomaly analysis framework for database systems. *Computers and Security* 53 (2015), 156 – 173.
- [122] VERSTREPEN, K., BHADURIY, K., CULE, B., AND GOETHALS, B. Collaborative filtering for binary, positiveonly data. *SIGKDD Explor. Newsl.* 19, 1 (Sept. 2017), 1–21.
- [123] VIEGAS, F., ROCHA, L., GONÇALVES, M., MOURÃO, F., SÁ, G., SALLES, T., ANDRADE, G., AND SANDIN, I. A genetic programming approach for feature selection in highly dimensional skewed data. *Neurocomputing* 273 (2018), 554 – 569.
- [124] WASTL, R. Linear derivations for keys of a database relation schema. *Journal of Universal Computer Science* 4, 12 (1998), 883–897.
- [125] WASTL, R. On the number of keys of a relational database schema. *Journal of Universal Computer Science* 4, 5 (1998), 547–559.
- [126] WIENKE, S., MILLER, J., SCHULZ, M., AND MÜLLER, M. S. Development effort estimation in hpc. In *Proceedings of the International Conference for High Performance Computing, Networking, Storage and Analysis* (Piscataway, NJ, USA, 2016), SC '16, IEEE Press, pp. 10:1–10:12.
- [127] WILLE, R. *Formal Concept Analysis as Mathematical Theory of Concepts and Concept Hierarchies*. Springer Berlin Heidelberg, Berlin, Heidelberg, 2005, pp. 1–33.
- [128] WORLAND, P. B. An efficient algorithm for 3nf determination. *Information Sciences* 167, 1 (2004), 177 – 192.



- [129] YANG, X., WANG, Z., XUE, J., AND ZHOU, Y. The reliability wall for exascale supercomputing. *IEEE Transactions on Computers* 61, 6 (June 2012), 767–779.
- [130] YANG, Z., AND CAI, Z. Detecting abnormal profiles in collaborative filtering recommender systems. *Journal of Intelligent Information Systems* (2016), 1–20.
- [131] YAO, H., HAMILTON, H. J., AND BUTZ, C. J. Fd mine: Discovering functional dependencies in a database using equivalences. In *ICDM* (2002), IEEE Computer Society, pp. 729–732.
- [132] YING, Y. The personalized recommendation algorithm based on item semantic similarity. In *Communication Systems and Information Technology* (Berlin, Heidelberg, 2011), M. Ma, Ed., Springer Berlin Heidelberg, pp. 999–1004.
- [133] YU, C. T., AND JOHNSON, D. T. On the complexity of finding the set of candidate keys for a given set of functional dependencies. *Inf. Process. Lett.* 5, 4 (1976), 100–101.
- [134] ZAMANI, H., AND SHAKERY, A. A language model-based framework for multi-publisher content-based recommender systems. *Information Retrieval Journal* (Online Feb 2018).
- [135] ZHANG, W., DU, Y. J., AND SONG, W. Recommender system with formal concept analysis. In *2015 International Conference on Information and Communications Technologies (ICT 2015)* (April 2015), pp. 1–6.
- [136] ZHANG, Y. Determining all candidate keys based on karnaugh map. *IEEE International Conference on Information Management, Innovation Management and Industrial Engineering 04* (2009), 226–229.
- [137] ZHOU, W., WEN, J., GAO, M., LIU, L., CAI, H., AND WANG, X. *A Shilling Attack Detection Method Based on SVM and Target Item*

Analysis in Collaborative Filtering Recommender Systems. Springer International Publishing, Cham, 2015, pp. 751–763.

- [138] ZOBEL, J. How reliable are the results of large-scale information retrieval experiments? In *Proceedings of the 21st Annual International ACM SIGIR Conference on Research and Development in Information Retrieval* (New York, NY, USA, 1998), SIGIR '98, ACM, pp. 307–314.
- [139] ZOU, C., ZHANG, D., WAN, J., HASSAN, M. M., AND LLORET, J. Using concept lattice for personalized recommendation system design. *IEEE Systems Journal* 11, 1 (March 2017), 305–314.

*-¿Qué te parece desto, Sancho? –Dijo Don Quijote–
¿Hay encantos que valgan contra la verdadera valentía?
Bien podrán los encantadores quitarme la ventura,
pero el esfuerzo y el ánimo, será imposible.*

El Ingenioso Hidalgo Don Quijote de la Mancha

Miguel de Cervantes



UNIVERSIDAD
DE MÁLAGA



UNIVERSIDAD
DE MÁLAGA



UNIVERSIDAD
DE MÁLAGA