UNIVERSIDAD DE GRANADA

Escuela Técnica Superior de Ingeniería Informática

Departamento de Ciencias de la Computación
e Inteligencia Artificial

ugr | Universidad de Granada

Modelos de Sistemas de Recuperación de Información Documental Basados en Información Lingüística Difusa

Memoria de Tesis presentada por

D. Antonio Gabriel López Herrera

para optar al grado de Doctor en Informática

Granada Enero de 2006

Editor: Editorial de la Universidad de Granada
Autor: Antonio Gabriel López Herrera
D.L.: Gr. 181 - 2005
ISBN: 84-338-3710-9

UNIVERSIDAD DE GRANADA

Escuela Técnica Superior de Ingeniería Informática

Departamento de Ciencias de la Computación
e Inteligencia Artificial

Modelos de Sistemas de Recuperación de Información Documental Basados en Información Lingüística Difusa

Memoria de Tesis presentada por

D. Antonio Gabriel López Herrera

para optar al grado de Doctor en Informática

Dr. D. Enrique Herrera Viedma
Director

Fdo. Enrique Herrera Viedma Fdo. Antonio Gabriel López Herrera

Granada Enero de 2006

La memoria titulada **Modelos de Sistemas de Recuperación de Información Documental Basados en Información Lingüística Difusa**, que presenta **D. Antonio Gabriel López Herrera** para optar al grado de Doctor en Informática, ha sido realizada en el **Departamento de Ciencias de la Computación e Inteligencia Artificial** de la Universidad de Granada bajo la dirección del Doctor **D. Enrique Herrera Viedma**.

Dr. D. Enrique Herrera Viedma Director	D. Antonio Gabriel López Herrera Doctorando

24 de Enero de 2006

Normalmente, cuando la gente escribe sus agradecimientos se “enrolla” y escribe “parrafadas” tremendas, en mi caso, la cosa es mucho más simple.

En primer lugar, quiero agradecer a Enrique Herrera, mi tutor, todos sus desvelos y dedicación durante tanto tiempo, a Paco Herrera, sus consejos y apoyo, y como no, a mis padres y mi hermano por su apoyo constante, y especialmente a Marisa, mi novia, por ilusionarse conmigo en los buenos momentos y “aguantarme” en los malos.

Por supuesto, no me puedo olvidar tampoco de nadie del grupo de investigación *Soft Computing y Sistemas de Información Inteligentes*, y en especial de Carlos Porcel, Jesús Alcalá y Sergio Alonso por su apoyo y ayuda.

MUCHAS GRACIAS A TODOS.

Índice general

Índice de Tablas

Índice de figuras

Capítulo 1

Planteamiento, Objetivos y Estructura de la Memoria

En un mundo globalizado que cambia rápidamente como es el de la actual *sociedad de la información y del conocimiento*, el estar permanentemente informado se ha convertido en una necesidad apremiante, en fuente de conocimiento y también de dinero. La proliferación de unidades y fuentes de información, tanto en el ámbito científico, profesional e incluso doméstico, la oleada reciente de subscripciones a servicios on-line de noticias, etc., pone de manifiesto la importancia que la sociedad da a estar permanentemente informada sobre temas que son de su interés. La "puesta al día" informativa permite tanto a la persona individual como a las organizaciones ser competitivas y tomar mejores decisiones.

Internet, la fuente de información más grande jamás conocida, es una de las principales fuentes de generación y transmisión de información. Uno de los problemas principales de Internet es el crecimiento constante y descontrolado de la información a la que los usuarios pueden acceder [64, 65]. Este crecimiento desmesurado está contribuyendo a que los usuarios tengan dificultades para encontrar la información que precisan de manera simple y eficiente. Por ello se hace necesario desarrollar sistemas que les ayu-

den a hacer frente a esta gran maraña de información en que se ha convertido Internet [59, 65]. Como consecuencia, las investigaciones en áreas relacionadas con la búsqueda o acceso a la información, ya sea en la Web o en cualquier otro sistema, han aumentado considerablemente en los últimos años [3, 4, 22, 27, 28, 35, 60, 80, 83].

Todas estas investigaciones están basadas en diferentes técnicas o filosofías de trabajo, pero se pueden englobar bajo un mismo concepto, el de Acceso a la Información (en inglés, *Information Seeking* [66]), término que describe cualquier proceso que hace posible filtrar la gran cantidad de información disponible y que el usuario únicamente acceda a información relevante para él.

En los últimos años estamos asistiendo a la aplicación creciente de distintas ciencias en el desarrollo de sistemas de acceso a la información con objeto de mejorarlos. En concreto, métodos, conceptos y técnicas de Inteligencia Artificial (**IA**) están siendo aplicados en los procesos de obtención de información con notable éxito [4, 60, 86], dando lugar a la aparición del concepto de *Web Intelligence* [96, 99], concepto que engloba a disciplinas tales como: *Semantic Web*, *Web Agents*, *Web Mining*, *Web Information Retrieval*, *Web Information Systems*, *Web-based Applications*, *Web Human-Media Engineering*, etc. Por tanto, el estudio y desarrollo de nuevas técnicas de acceso a la información basadas en Web Intelligence, se muestra como una línea de investigación muy activa.

De entre todos los tipos de sistemas de acceso a la información destacamos dos [5]:

- Los sistemas de acceso a la información basados en los métodos tradicionales de Recuperación de Información (RI) que se encargan de dar respuesta a necesi-

dades de información puntuales que puedan tener los usuarios. Estas necesidades quedan representadas como consultas que los usuarios introducen en el sistema y automáticamente obtienen una respuesta, de modo que los resultados que se van obteniendo dependen en gran medida de la habilidad que los usuarios tengan de expresar mediante consultas sus necesidades de información. Son los más extendidos y se conocen con el nombre de buscadores [4] que se centran en obtener información relevante para los usuarios. Su actividad se desarrolla on-line, por lo que el sistema no dispone de ningún tipo de conocimiento a priori sobre los usuarios.

- Sistemas de acceso a la información basados en técnicas de Filtrado de Información (FI). El Filtrado de Información es un término usado para describir toda una variedad de procesos involucrados en la entrega de información exclusivamente a quiénes la necesitan. Por tanto, estos sistemas evalúan y filtran la gran cantidad de información disponible para los usuarios y así ayudarles en sus procesos de acceso a dicha información. En este caso, el sistema intenta dar respuesta a necesidades de los usuarios más persistentes en el tiempo, y en lugar de representar dichas necesidades mediante consultas puntuales, éstas son deducidas a partir de Perfiles de Usuario. Observamos que este tipo de sistemas sí tienen un conocimiento sobre los usuarios, almacenando mediante perfiles las preferencias o características de los mismos, por lo que en este caso la forma de trabajo es off-line. Los sistemas anteriores trabajan buscando información relevante mientras que los sistemas de FI persiguen satisfacer las necesidades de los usuarios recomendando información personalizada, de ahí que se hayan popularizado bastante con el nombre de Sistemas de Recomendaciones (SR) [75].

En cualquier caso, ambos tienen el objetivo de ayudar al usuario a satisfacer sus necesidades de información. En este sentido, Belkin y Croft [5] determinaron que el FI y la RI constituyen las dos caras de una misma moneda que, trabajando en estrecha relación, consiguen ayudar a los usuarios en la obtención de la información que necesitan para lograr sus objetivos. De hecho, usando sistemas de filtrado de información, podemos depurar la información seleccionada por los sistemas de recuperación de información, de manera que la información mostrada finalmente a los usuarios se adapte lo mejor posible a sus necesidades.

Por otro lado, nos enfrentamos al problema de disponer de una gran variedad de posibilidades a la hora de representar y evaluar la información [4, 47]. El problema se agrava aún más en los procesos en los que intervienen los usuarios, que muchas veces no son capaces de representar sus necesidades o preferencias de información de una forma adecuada, sino más bien de forma subjetiva, imprecisa o vaga [74, 93]. Se hace, pues, necesario el uso de técnicas para el manejo de información subjetiva, imprecisa y cualitativa como son las técnicas de Modelado Lingüístico Difuso para crear un entorno de trabajo flexible [7, 30, 44, 47, 97].

De entre todos los procesos en el acceso a la información, en esta memoria solo nos vamos a centrar en la recuperación de información.

La RI se puede definir como el problema de la selección de información en respuesta a consultas o demandas de información por parte de un usuario [4, 80, 83, 90]. Los Sistemas de Recuperación de Información (SRI) son una clase de sistemas de información que tratan con bases de datos compuestas por documentos y procesan las consultas de

los usuarios permitiéndoles acceder a la información relevante en un intervalo de tiempo apropiado. Estas consultas son sentencias formales mediante las cuales el usuario expresa sus necesidades de información y suelen venir expresadas por medio de un lenguaje de consulta.

La mayoría de los SRI comerciales se basan en el modelo Booleano [90], y presentan limitaciones para manejar la información vaga, imprecisa y subjetiva que aparece tanto en la interacción con los usuarios como en los procesos de búsqueda.

Para resolver este problema se están desarrollando SRI basados en Técnicas de Conjuntos Difusos [8, 12, 17, 50, 51, 70, 62]. Dentro de estos, los más flexibles y los que más facilidad de interacción usuario-sistema ofrecen son los SRI difusos basados en información lingüística difusa [6, 12, 13, 61]. Estos son diseñados usando el concepto de variable lingüística [97] para representar mejor la información cualitativa y cuentan con lenguajes de consultas ponderados lingüísticos que mejoran la interacción SRI-usuario. Estos lenguajes de consulta, por un lado, incrementan las posibilidades de expresión de los usuarios porque con ellos es posible asignar pesos a los términos de las consultas indicando importancia relativa o umbrales de satisfacción, y por otro, facilitan a los usuarios la expresión de sus necesidades de información porque pueden expresar los pesos mediante valores lingüísticos más propios del lenguaje humano. Se han propuesto diferentes modelos de SRI lingüísticos usando una aproximación lingüística difusa ordinal que facilita la expresión y el procesamiento de los pesos de las consultas [50, 51, 52, 53]. Las principales limitaciones de los anteriores SRI lingüísticos son: i) la pérdida de precisión e información en los procesos de cómputo, ii) el uso de operadores de agregación y funciones de evaluación de bajo rendimiento y iii) la imposibilidad de

tratar con información lingüística no balanceada. Los anteriores SRI, para establecer los pesos en los términos de las consultas, suelen asumir un conjunto de etiquetas simétrica y uniformemente distribuidos alrededor de la etiqueta central, fijando el mismo nivel de discriminación a ambos lados de ésta. Usando información lingüística no balanceada, el usuario podría aumentar el grado de discriminación de uno de estos lados.

1.1. Objetivos

El objetivo del trabajo desarrollado en la presente memoria es profundizar en la mejora de los SRI diseñados usando técnicas de modelado lingüístico difuso, de cara a mejorar tanto la interacción usuario-SRI, como los procesos de evaluación de consultas que realizan dichos sistemas. Para ello, aplicaremos un conocido de representación de información difuso, el modelo lingüístico difuso 2-tupla [46], y propondremos nuevos mecanismos de evaluación de consultas y definiremos un nuevo modelo para manejar información lingüística no balanceada con el cual diseñaremos un SRI lingüístico no balanceado que mejore las posibilidades de expresión.

Este objetivo global se desglosa en los siguientes subobjetivos:

1. Revisión de los SRI lingüísticos y técnicas de modelado de información.
2. Diseñaremos nuevos modelos de SRI usando diferentes aproximaciones lingüísticas:
   a) modelo de SRI lingüístico 2-tupla,
   b) modelo de SRI lingüístico no-balanceada.
3. Desarrollaremos técnicas para mejorar la evaluación de las consultas de usuario:

a) nuevas funciones de evaluación (matching functions) para interpretar las distintas semánticas (umbral, cuantitativa, ...),

b) nuevos operadores de agregación de información lingüística más flexibles.

4. Evaluación de las distintas propuestas con respecto a otros SRI propuestos en la literatura.

1.2. Estructura de la Memoria

Así, la presente memoria se divide en seis capítulos y dos anexos y se estructura como sigue:

Capítulo 2: se hace una breve introducción a los sistemas de recuperación de información, con el fin de acercar al lector al problema objeto de estudio.

Capítulo 3: introducimos los conceptos y las herramientas de modelado lingüístico que utilizaremos a lo largo de la memoria, que no son otras que:

- conjuntos difusos, variable lingüística, ...
- enfoque ordinal de representación de información lingüística, junto con sus operadores de agregación,
- enfoque 2-tupla de representación de información lingüística, junto con algunos de agregación, y
- metodología para agregar información multigranular.

Y revisaremos algunos de los SRI lingüísticos propuestos en la literatura.

Capítulo 4: En este capítulo abordaremos los objetivos: 2.a, 3.a, 3.b y 4. Es decir, propondremos un modelo lingüístico de recuperación de información documen-

tal basado en el modelo de representación de información 2-tupla, también este sistema incorporará una nueva interpretación de una semántica de umbral, así como unos nuevos operadores de agregación mucho más flexibles, por último, este sistema será evaluado y comparado con otros propuestos en la literatura.

Capítulo 5: El resto de objetivos, 2.b, 3.b y 4, serán cubiertos en este capítulo. Concretamente, propondremos un modelo lingüístico de recuperación de información documental basado igualmente en el modelo 2-tupla, pero ahora, permitiendo usar información lingüística no balanceada, también propondremos el modelo de cómputo asociado para manejar este tipo de información, e igualmente procederemos a evaluarlo.

Capítulo 6: Algunos comentarios, incluyendo conclusiones finales y trabajos futuros serán esbozados.

Finalmente, en el anexo A, describiremos el software desarrollado, el cual implementa en un mismo sistema, todas las ideas desarrolladas teóricamente.

Capítulo 2

Introducción a los Sistemas de Recuperación de Información

En este capítulo vamos a repasar los conceptos básicos de la Recuperación de Información, presentaremos los Sistemas de Recuperación de Información, analizando sus componentes principales, estudiaremos los distintos modelos de recuperación que se han propuesto en la literatura, profundizando en el modelo difuso de RI que será el que empleemos en esta memoria.

2.1. Introducción

Los avances tecnológicos de los últimos cincuenta años han provocado un aumento exponencial de la información y una mejora de su difusión. Hoy nos hallamos inmersos en la revolución de la información, cada vez tenemos más información disponible y mayores posibilidades para accederla. El proceso de digitalización de los documentos así como el desarrollo de nuevas tecnologías de la información tanto en su creación, como en su distribución, como en su acceso, son dos claros ejemplos de la revolución de la información, lo cual ha permitido su acceso y uso por un número ilimitado de usuarios.

Además, hay que tener en cuenta que el uso masivo de las tecnologías y de los ordenadores no se reduce a la producción editorial, sino que está presente en todos los ámbitos de la vida, sobre todo en el trabajo, y hasta en el hogar donde cada vez es mayor el número de personas que no sólo tienen ordenador sino que poseen equipos multimedia. A ello habría que sumar la distribución de información mediante las llamadas "autopistas de la información", la proliferación de las conexiones de banda ancha y el coste cada vez menor de los medios de almacenamiento. Todo ello nos sitúa dentro de un entorno en desarrollo de información electrónica a la que se puede acceder por medios automáticos. Otro aspecto que tenemos que considerar es la diversificación de los medios, que trae consigo una mayor cantidad de información no normalizada, imagen, sonido, texto, etc.

La Recuperación de Información (RI) se puede definir como el problema de la selección de información, depositada en un medio de almacenamiento, en respuesta a consultas realizadas por un usuario [4, 80, 83, 90].

Los Sistemas de Recuperación de Información (SRI) son una clase de sistemas de información que tratan con bases de datos compuestas por documentos y procesan las consultas de los usuarios permitiéndoles acceder a la información relevante en un intervalo de tiempo apropiado (véase la Figura 2.1). Estas consultas son sentencias formales mediante las cuales el usuario expresa sus necesidades de información, formuladas usando un lenguaje de consulta. Estos sistemas fueron originalmente desarrollados en la década de los años 40 con la idea de auxiliar a los gestores de la documentación científica.

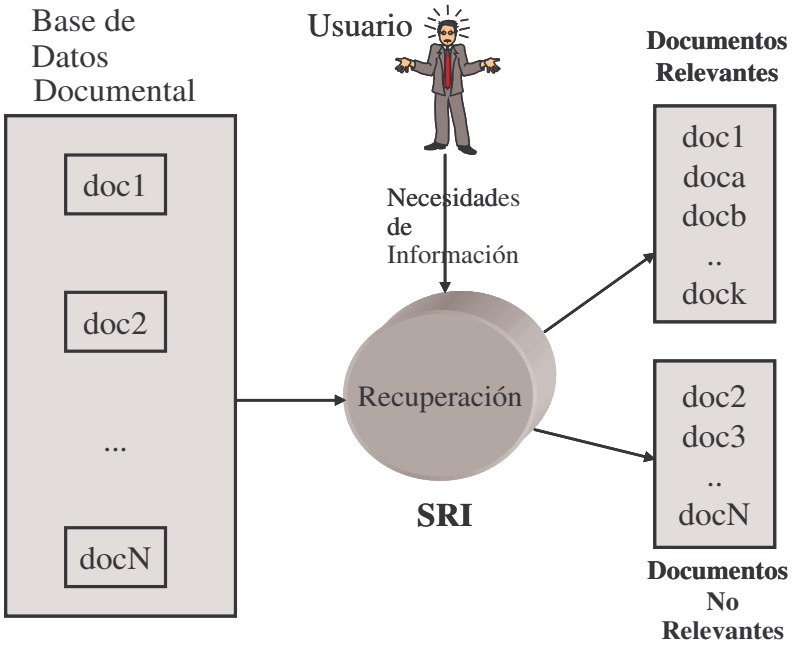

Figura 2.1: Proceso de recuperación de información.

Un SRI debe soportar una serie de operaciones básicas sobre los documentos almacenados, como son: introducción de nuevos documentos, modificación de los que ya estén almacenados y eliminación de los mismos. Debemos también contar con algún método de localización de los documentos (o con varios, generalmente) para presentárselos posteriormente al usuario. Este proceso se resume gráficamente en la Figura 2.2. Los SRI implementan estas operaciones de varias formas distintas, lo que provoca una amplia diversidad en los mismos. Por tanto, para estudiarlos es necesario establecer en primer lugar una clasificación de estos sistemas. Para ello, veremos a continuación cuáles son los componentes principales de un SRI.

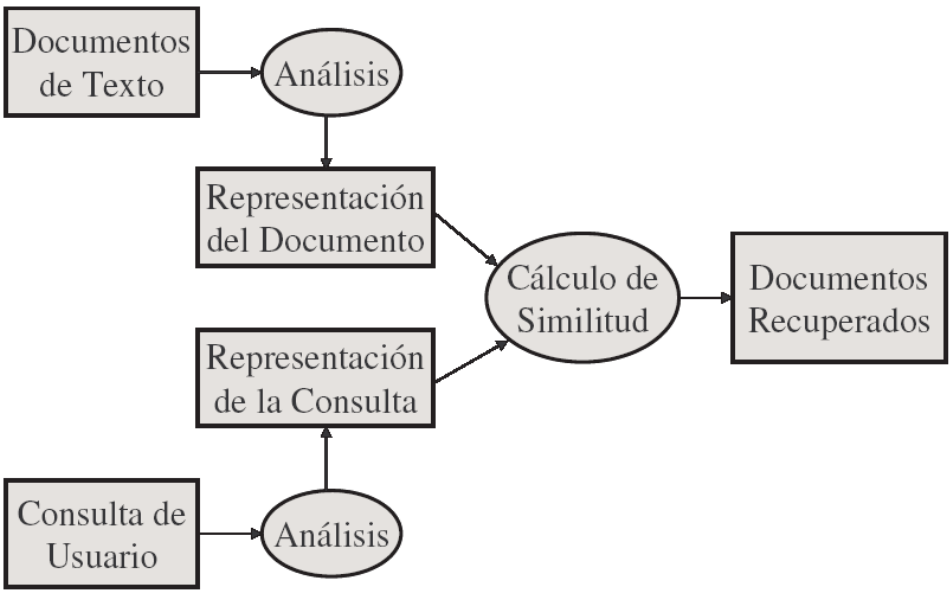

Figura 2.2: Operaciones para la recuperación de documentos.

2.2. Componentes de los Sistemas de Recuperación de Información

Un SRI está compuesto por tres componentes principales: la *base de datos documental*, el *subsistema de consultas* y el *mecanismo de emparejamiento o evaluación* (Figura 2.3). Las tres secciones siguientes están dedicadas a estudiar la composición de cada uno de ellos.

2.2.1. La Base de Datos Documental

Un documento es un conjunto de datos, de naturaleza tradicionalmente textual, aunque la evolución tecnológica ha propiciado la aparición de documentos multimedia, incorporándose al texto fotografías, ilustraciones gráficas, vídeos animados, audio, etc.

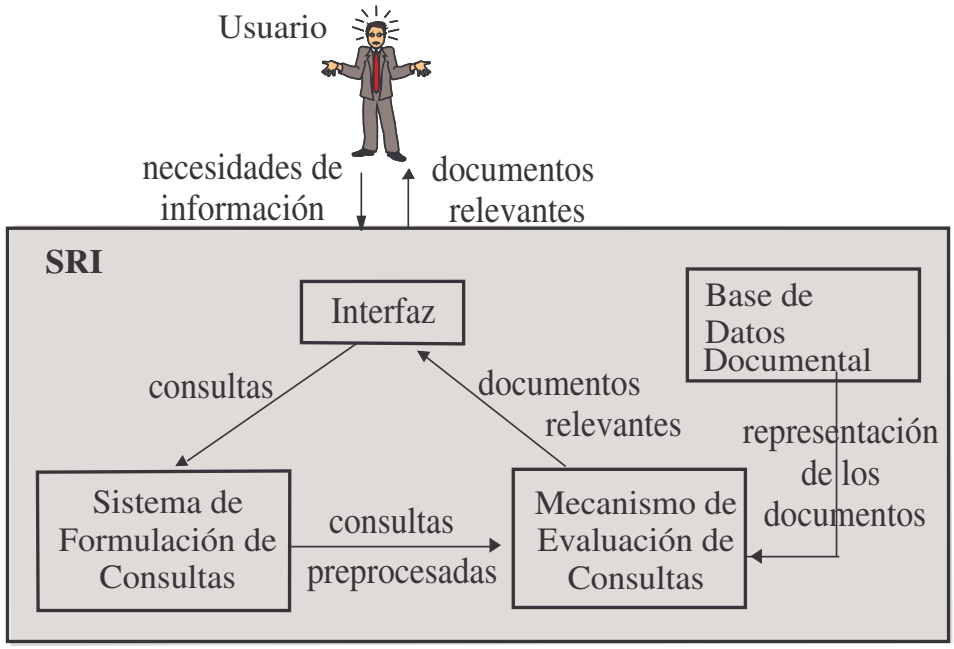

Figura 2.3: Componentes básicos de un sistema de recuperación de información.

Aunque la variedad en cuanto a documentos se refiere, está aumentando tanto en soportes como en el carácter de su contenido, nosotros nos vamos a centrar en los que tienen naturaleza únicamente textual.

Estos documentos no se almacenan directamente en el SRI, sino que se preprocesan y se representan por un conjunto de elementos llamados *descriptores*. Por tanto, un documento se compondrá de una serie de descriptores.

Desde un punto de vista matemático, la base de datos es una tabla o matriz en la que cada fila representa a un documento y cada columna indica la presencia, o no, de un determinado descriptor en el documento correspondiente. En principio, en cada fila aparecen "unos" en las columnas relativas a los descriptores asignados al documento y

"ceros" en las restantes. Así, cada documento estará representado por un vector de ceros y unos [90]. Podemos pensar que esta representación se podría mejorar introduciendo información numérica sobre la asignación de un descriptor al documento en lugar de simplemente 0 y 1. Como veremos a continuación, esta operación se tendría que hacer teniendo en cuenta toda la base documental y el universo de conceptos. La información numérica de la asignación de un concepto a un documento puede tener diferentes significados dependiendo del modelo de recuperación que se trate. Por ejemplo, en el modelo de Espacio Vectorial [83], que estudiaremos en la Sección 2.3.2, puede considerarse como el grado en el que ese descriptor describe el documento; mientras que en el modelo Probabilístico [9] (Sección 2.3.3), se considera como la probabilidad de que el documento sea relevante para ese descriptor.

Podemos considerar una base documental $\mathcal{D}$, compuesta por documentos d_i, indizada por un conjunto de términos, $\mathcal{T}$, formado por n términos t_j, en la que cada documento d_i contiene un número no especificado de términos de indización t_j. De esta forma, sería posible representar cada documento como un vector (o conjunto, aplicando la terminología del modelo booleano, Sección 2.3.1) perteneciente a un espacio n-dimensional, siendo n el número de términos de indización que forman el conjunto $\mathcal{T}$:

$$d_i = (t_{i1}, t_{i2}, t_{i3}, \ldots, t_{in})$$

donde cada uno de los elementos t_{ij} de este vector puede representar la presencia o ausencia del término t_j en el documento d_i en la indización binaria, la relevancia del término t_j en el documento d_i en el modelo de espacio vectorial, o la probabilidad de que el documento d_i sea relevante al término t_j en el modelo probabilístico.

La indización (proceso de construcción de los vectores documentales) puede realizarse de forma manual o automática. En este último caso, la base de datos documental comprende un módulo llamado módulo indizador que se encarga de generar automáticamente la representación de los documentos extrayendo los contenidos de información de los mismos. La labor del módulo indizador consistirá en asociar automáticamente una representación a cada documento en función de los contenidos de información de éste, es decir, determinar los pesos de cada término en el vector documental. Su función de indización o ponderación será:

$$\mathcal{F} : \mathcal{D} \times \mathcal{T} \longrightarrow [0, 1]$$

La representación de cada vector tendrá n componentes, de los cuales los que estén referenciados en el documento tendrán un valor diferente de 0, mientras que los que no estén referenciados tendrán un valor nulo o 0. Es importante señalar que la indización juega un papel fundamental en la calidad de la recuperación, siendo crucial la elección apropiada del método de indización.

De este modo, para obtener estas representaciones se aplica un proceso de "construcción de la base documental". Para ello, solemos partir de una información mucho menos específica, es decir, del estado puro del documento (información textual). Partiendo de esta información, el sistema realizará un conjunto de operaciones que permitirán obtener la base de datos documental [4, 83].

Dichas operaciones están representadas gráficamente en la Figura 2.4.

Los documentos de tipo textual se pueden representar bien por una componente estructurada en campos (título, autor, resumen, palabras clave, ...) o bien por una

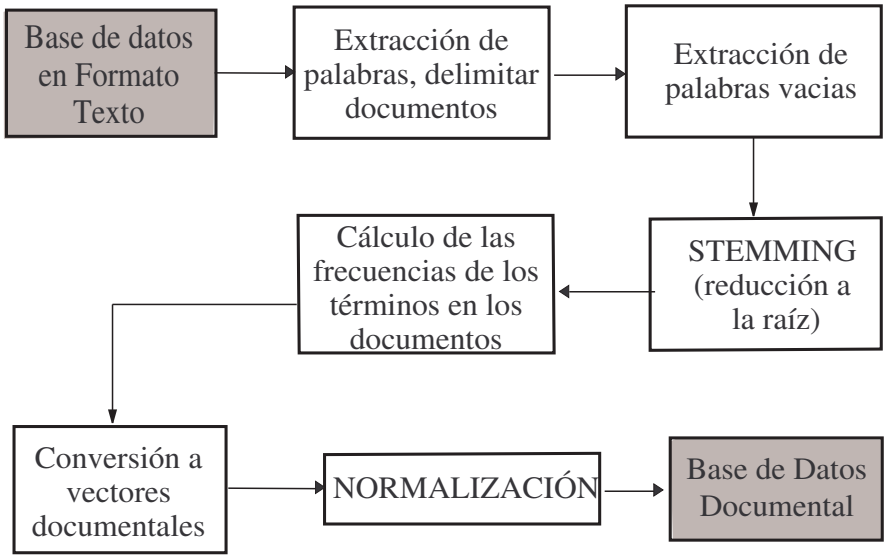

Figura 2.4: Proceso documental.

componente no estructurada, es decir, el texto literal. La representación textual de cada documento se basará normalmente en los términos de indización (o descriptores, que pueden ser tanto palabras individuales como asociaciones de éstas). Para representar la parte no estructural, el primer paso para la construcción de la base documental consiste en extraer los términos del texto del documento.

A continuación, analizaremos más detenidamente el proceso que siguen los documentos para pasar a formar parte de la base de datos documental.

Preprocesamiento

El primer paso, incluso anterior a los que hemos nombrado antes, es el denominado "preprocesamiento", el cual consiste en eliminar aquellos fragmentos de texto que no tienen nada que ver con el documento a tratar. Se trata, por tanto, de un análisis de patrones léxicos en el flujo del texto. Como resultado de este preprocesamiento

obtendremos los documentos delimitados y sin cabeceras informativas que no nos sean útiles.

Vectorización

En este momento, contamos con todos los términos existentes en todos los documentos que forman la base de datos documental. La siguiente pregunta es: ¿qué términos son los que usaremos realmente para indexar un documento?. La base para responder a esta pregunta, nos la da, por un lado, el trabajo que llevó a cabo Lunh [83, 90], quién planteaba que la frecuencia de aparición de una palabra en un texto determinaba su importancia en él, sugiriendo que dichas frecuencias pueden ser utilizadas para extraer palabras con objeto de resumir el contenido de un documento. Por otro lado, está la ley de Zipf [83, 90, 102], que establece que si obtenemos la frecuencia de aparición, f, de cada palabra de un texto y la ordenamos decrecientemente, siendo p la posición que ocupa en dicha ordenación, se cumple que $f \cdot p \simeq c$, donde c es una constante.

Si se representa gráficamente esta curva (p en el eje X, y f en el Y), se obtiene una hipérbola, en la cual se pueden establecer dos límites en cuanto a p se refiere (véase la Figura 2.5): todas las palabras que excedan el límite superior, se considerarán muy comunes (haciendo una búsqueda por ellas podríamos recuperar casi todos los documentos), y todas las que estén por debajo del límite inferior, muy raras. Las palabras con frecuencias intermedias, es decir, las que queden dentro de ambos límites serán las que tengan una mayor capacidad (poder de resolución) para discriminar el contenido de un texto y, por tanto, las que deban ser usadas. El problema radica en establecer los dos límites anteriores, porque, tal y como dicen Salton y McGill en [83], la eliminación de palabras con frecuencias muy altas puede provocar una reducción de la exhaustividad,

ya que el uso de conceptos generales es útil a la hora de recuperar muchos documentos relevantes. Por el contrario, el descartar términos con una frecuencia baja, produce pérdidas en la precisión. Intentando paliar estos problemas, Pao ofrece un método para calcular automáticamente el límite inferior [73].

Otro aspecto a tener en cuenta a la hora de seleccionar los términos consiste en eliminar las palabras vacías de significado, como pueden ser artículos, preposiciones, conjunciones, incluso en algunos casos, se pueden calificar así algunos verbos, adverbios y adjetivos [4].

Por tanto, estas palabras vacías de significado no nos sirven como términos de indexación, ya que, por un lado son muy frecuentes, y por otro no representan correctamente el contenido del documento [60]. La acción habitual que se lleva a cabo con ellas es su eliminación del texto, proceso que se conoce como *eliminación de palabras vacías* (stopwords[1] en inglés), y se pone en práctica mediante la comparación de cada palabra del texto con un diccionario que contiene la lista de palabras no aptas para la indexación (tanto en [90] como en [37] se presentan dos listas completas de palabras vacías).

Llegados a este momento, tenemos todas las palabras que nos interesan para la indización correcta del documento, pero aún así necesitamos ser un poco más parcos con nuestra información para mejorar el rendimiento del SRI. El siguiente paso consiste en ofrecer al usuario la posibilidad de encontrar las variantes morfológicas de los términos de búsqueda. Procederemos por tanto a la reducción a la raíz de las palabras restantes.

[1] Hay que señalar que este conjunto de palabras vacías dependerá del lenguaje en el que se esté realizando el proceso de indización. Así por ejemplo, el conjunto de artículos del español y del inglés son diferentes. En [1] podemos encontrar listados de palabras vacías para una serie de idiomas.

Este proceso se conoce como *stemming* y se utiliza también para reducir el tamaño de los ficheros índice. Almacenando sólo las raíces de los términos en cuestión, se puede llegar a reducir su dimensión hasta un 50 %. La reducción de los términos puede realizarse bien durante la indización o bien en la propia búsqueda. La primera variante presenta la ventaja de ser más eficiente y ahorrar espacio, pero tiene la desventaja de perder información sobre los términos completos.

Existen cuatro variedades automáticas de stemming [38] que analizaremos a continuación:

- *Eliminación de afijos*: trata de eliminar los prefijos y/o los sufijos de los términos, quedando la raíz. Este método es el más utilizado. Uno de los algoritmos de este tipo más conocidos y empleados es el de Porter [76].

- *Variedad de sucesores*: basándose en la frecuencia de las secuencias de letras en un texto.

- *N-gramas*: combinación de términos basados en el número de diagramas o ngramas que comparten.

- *Búsqueda en tabla*: en la que están contenidos los términos y sus correspondientes raíces.

Sólo nos resta decir sobre este proceso que el stemming dejará de ser correcto tanto si las palabras se recortan en exceso como si no se recortan lo suficiente, ya que provocaría *ruido* (recuperación de documentos no relevantes) o *silencio* (la no recuperación de documentos relevantes).

La última etapa del proceso de selección pasa por determinar la importancia de cada palabra (término) en el documento, de tal forma que, si es lo "suficientemente" importante, se escogerá para ser incluida en el conjunto de términos final. El cálculo de la importancia de cada término se conoce como *ponderación del término*.

¿Cómo se mide esa importancia?. Un primer enfoque se basa en contar las ocurrencias de cada término en un documento, medida que se denomina frecuencia del término i-ésimo en el documento j-ésimo, y se nota como $tf_{i,j}$. Una segunda medida de la importancia del término es la conocida como *frecuencia documental inversa* de un término en la colección, conocida normalmente por sus siglas en inglés: *idf* (inverse document frequency), que inicialmente ideó Luhn [85] y que posteriormente formalizó Salton [80, 83], y que responde a la siguiente expresión:

$$idf_i = \log(\frac{N}{n_i}) + 1 \tag{2.1}$$

donde N es el número de documentos de la colección, y n_i el número de documentos donde se menciona al término i-ésimo. Como se puede observar, el valor idf_i decrece conforme n_i crece, variando desde $log(N) + 1$ cuando n_i es 1, a 1 cuando n_i toma el valor N. Por tanto, cuantas menos veces aparezca un término en la colección, más alto será su *idf* [60], dando así una forma de medir la calidad global del término en toda la colección. El hecho de introducir un logaritmo se justifica para suavizar el crecimiento del tamaño de la colección.

Lo ideal sería combinar ambas medidas anteriores utilizando un esquema de ponderación que permita identificar a los términos que aparecen con frecuencias altas en varios documentos individuales, y a la vez, que se hayan observado en contadas oca-

siones en la colección completa. Estos son los términos que tendrán una capacidad de discriminación mayor con respecto a los documentos en los que aparecen. O lo que es lo mismo, calcular un peso que fuera proporcional a la frecuencia del término i-ésimo en el documento j-ésimo, e inversamente proporcional al número de documentos de la colección completa en los que aparece ese término. Así, el peso final asignado al término i-ésimo en el documento j-ésimo, que notaremos como $tf \cdot idf$, corresponde al producto:

$$tf_{i,j} \cdot idf_i$$

En este caso, la importancia crece con respecto a la frecuencia del término en el documento y disminuye con respecto al número de documentos que lo contienen [60]. Cuanto más alto sea este valor, mejor será el término desde el punto de vista de la indexación. Existen otras medidas como son el valor de discriminación del término, y la relación señal/ruido [60, 83], que se plantean como alternativas totalmente viables al $tf \cdot idf$.

Podemos indizar un libro, artículo, tesis, disertación, etc. y, lo que es más importante, esto se puede hacer usando procesamiento automático, siempre y cuando se apliquen y respeten ciertas reglas.

Una vez que hemos obtenido todos los términos con mayor poder discriminatorio, es decir, los más representativos y cargados de información, procederemos a la vectorización. Este proceso consiste en la construcción de vectores con el tamaño de los términos significativos que han quedado. Es decir, un documento d_i se identificará mediante una colección de términos $t_{i1}, t_{i2}, t_{i3}, \ldots, t_{it}$, donde t_{ij} representa el peso, o importancia, del término j en el documento i, como hemos visto al principio de la Sección

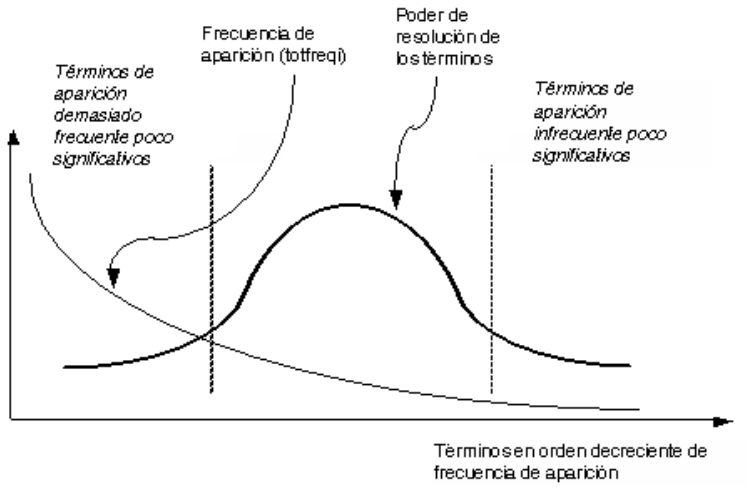

Figura 2.5: Representación gráfica de la frecuencia de los términos ordenados según su posición en la ordenación: ley de Zipf.

2.2.1. Por "*término*" entendemos una especie de identificador de contenido, como una palabra extraída de un documento, de una frase, o una entrada de un tesauro. Por tanto, una base documental podría representarse como una ordenación, o matriz, de términos donde cada fila de la matriz representa un documento y cada columna representa la asignación de un término específico a los documentos en cuestión, como en la Figura 2.6.

A continuación, se construyen los vectores con el tamaño de los términos significativos escogidos finalmente y se les asigna un peso usando la función de ponderación.

2.2.2. El Subsistema de Consulta

Este subsistema está compuesto por la interfaz que permite al usuario formular sus consultas y por un analizador sintáctico que toma la consulta escrita por el usuario y la desglosa en sus partes integrantes. Para llevar a cabo esta tarea, incluye un lenguaje

	t1	t2	....	tt
doc1	t11	t12	....	t1t
doc2	t21	t22	....	t2t
....	....	....	....	....
docn	tn1	tn2	....	tnt

Figura 2.6: Representación matemática de la base documental.

de consulta que recoge todas las reglas para generar consultas apropiadas. La interfaz ofrecerá facilidades al usuario a la hora de formular su consulta, ya que éste no tiene por qué saber exactamente el funcionamiento tanto externo como interno del sistema. También se ocupará de mostrar al usuario el resultado de su búsqueda, una vez procesada su consulta. En muchas ocasiones los usuarios de SRI realizan sus peticiones basándose en la estructura de consultas Booleanas (con operadores Booleanos, es decir, AND, OR, NOT). Cada uno de los elementos básicos de la consulta puede ser un término (descriptor o concepto).

Como hemos comentado, la consulta que proporcione el usuario no puede procesarse directamente en su forma original, ha de recibir un tratamiento previo que consiste en desglosar la consulta en sus componentes básicos, además de comprobar que corresponde con el formato que se espera de ella (es decir, que su composición es correcta y se ajusta con las reglas del lenguaje de consulta). Esta comprobación se podrá llevar a cabo tanto a priori como a posteriori. Si se realiza a priori, el sistema directamente

no permite al usuario ejecutar su consulta hasta que no esté en el formato correspondiente. Si la comprobación se realiza a posteriori, el sistema devolverá al usuario un mensaje de error o un resultado incongruente. El análisis de la consulta se llevará a cabo mediante un analizador sintáctico, que determinará si la consulta es correcta o no y la desglosará en sus componentes. Después de esta partición, se podrá llevar a cabo el proceso de stemming para obtener las raíces de los términos de consulta. Finalmente la consulta se indizará o vectorizará y será enviada al mecanismo de evaluación para que éste determine qué documentos se consideran relevantes a la consulta proporcionada por el usuario.

2.2.3. El Subsistema de Evaluación

Llegados a este punto, tenemos una representación del contenido de los documentos en nuestra base documental y también una representación de las consultas que queremos realizar proveniente del subsistema de consulta. Lo que nos queda por resolver es la selección de los documentos que se consideran relevantes, de entre los documentos que forman la base documental, de acuerdo con los criterios de nuestra consulta. De esto precisamente se encargará el subsistema de evaluación. Este subsistema calcula el grado en el que las representaciones de los documentos satisfacen los requisitos expresados en la consulta y recupera aquellos documentos que son relevantes a la misma. Este grado es lo que se denomina **RSV** (Retrieval Status Value en inglés). Principalmente, existen dos modalidades de evaluación: sistemas que emparejan los documentos individualmente con la consulta, uno por uno; y otros que los emparejan en su conjunto [38].

Dedicaremos la sección siguiente a analizar los modelos de RI más conocidos.

2.3. Clasificación de los Sistemas de Recuperación de Información

Existen varios modelos o técnicas de RI y, como en todo, cada uno tiene sus ventajas e inconvenientes. En esta sección haremos una introducción a varios de los modelos existentes y analizaremos las componentes que los forman. Los principales modelos clásicos de recuperación de información son: modelo Booleano, modelo Espacio Vectorial, modelo Probabilístico y modelo Booleano extendido o modelo Difuso.

2.3.1. Modelo Booleano

Este modelo se basa en la teoría del álgebra de Boole. Se denomina Algebra de Boole o Algebra Booleana a las reglas algebraicas, basadas en la teoría de conjuntos, para manejar ecuaciones de lógica matemática. La lógica matemática trata con proposiciones, elementos de circuitos de dos estados, etc., asociados por medio de operadores como **AND**, **OR**, **NOT**, **IF...THEN**. Por tanto, permite cálculos y demostraciones como cualquier parte de las matemáticas, además de posibilitar la codificación de la información en el ámbito computacional. Se denomina así en honor de George Boole, famoso matemático, que la introdujo en 1847. A continuación introduciremos las componentes principales de este modelo [90].

Indización de Documentos en el Modelo Booleano

Dentro de un sistema Booleano, los documentos se encuentran representados por conjuntos de palabras clave (términos). La indización se realiza asociando un peso binario a cada término del índice: 0 si el término no aparece en el documento y 1 si aparece aunque sea una sola vez. Las búsquedas consisten en expresiones de palabras

claves conectadas con algún/os operador/es lógico/s (AND, OR y NOT). El grado de similitud entre un documento y una consulta será también binario y un documento será relevante cuando su grado de similitud sea igual a 1, de lo contrario el documento no tendrá ninguna relevancia en cuanto a la consulta. Por tanto, en el caso de los SRI Booleanos, la función de indización quedaría así:

$$\mathcal{F} : \mathcal{D} \times \mathcal{T} \longrightarrow \{0, 1\}$$

El Subsistema de Consulta en el Modelo Booleano

Como hemos comentado, las consultas en este modelo se compondrán de expresiones Booleanas que comprenden el conjunto de términos $\mathcal{T}$ y los operadores Booleanos AND, OR y NOT. Un ejemplo de este tipo de consultas sería:

$$(t_1 \mathbf{AND} t_2) \mathbf{OR} (t_2 \mathbf{AND\ NOT} t_9)$$

Que gráficamente puede visualizarse en forma de árbol como muestra la Figura 2.7.

Cuando se ejecute la consulta, el subsistema de consulta extraerá el RSV de cada documento y decidirá qué conjunto de documentos es el que se considera relevante para dicha consulta. En este modelo, esta operación es muy sencilla ya que no existe gradación de relevancia (el documento es totalmente relevante a la consulta o no lo es en absoluto). Por tanto, los valores del RSV serán 0 o 1 y formarán el conjunto de documentos recuperados aquellos que tengan el RSV igual a 1.

El Subsistema de Evaluación en el Modelo Booleano

El trabajo del subsistema de evaluación de este modelo consiste en emparejar la consulta $\mathcal{Q}$ con la representación de los documentos de la base documental para obtener, de este modo, el RSV de cada uno de ellos. Para obtener el conjunto de documentos

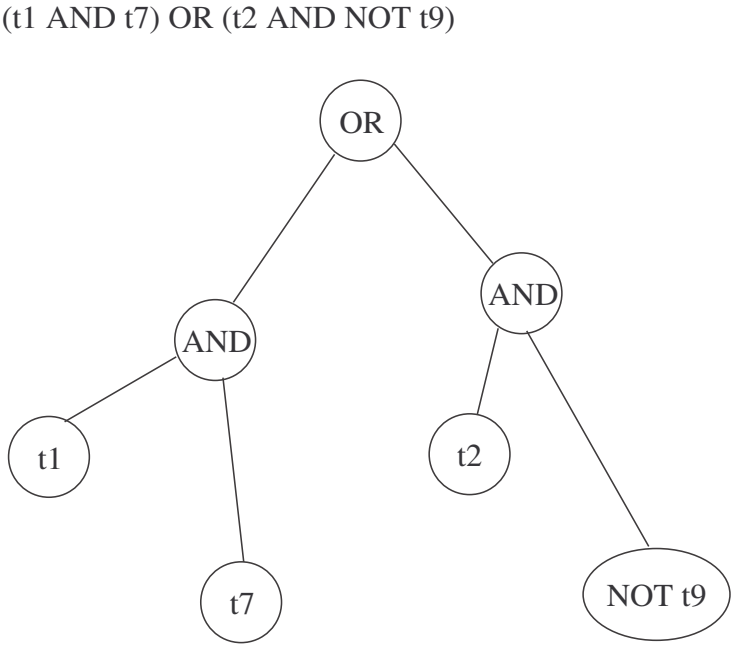

Figura 2.7: Ejemplo de consulta en el modelo Booleano.

relevantes, se recorrerá el árbol de la consulta de abajo a arriba, es decir, de las hojas a la raíz. Para ello, nos situamos en una hoja y determinamos el conjunto de documentos relevantes para el término situado en ella, es decir, aquellos que tienen dicho término (o que no lo tengan en caso de negación). Posteriormente, vamos subiendo en el árbol aplicando la operación correspondiente en cada nodo para obtener el conjunto de documentos asociado (intersección de conjuntos para el caso del *AND*, y unión de conjuntos con el *OR*). Finalmente, el conjunto de documentos devuelto por el sistema es el contenido en el nodo raíz. La Figura 2.8 muestra un ejemplo de evaluación en este modelo.

La ventaja del modelo Booleano es que es un modelo muy simple, basado en el Álgebra de Boole, lo que le da un marco teórico sólido. Su principal desventaja es el

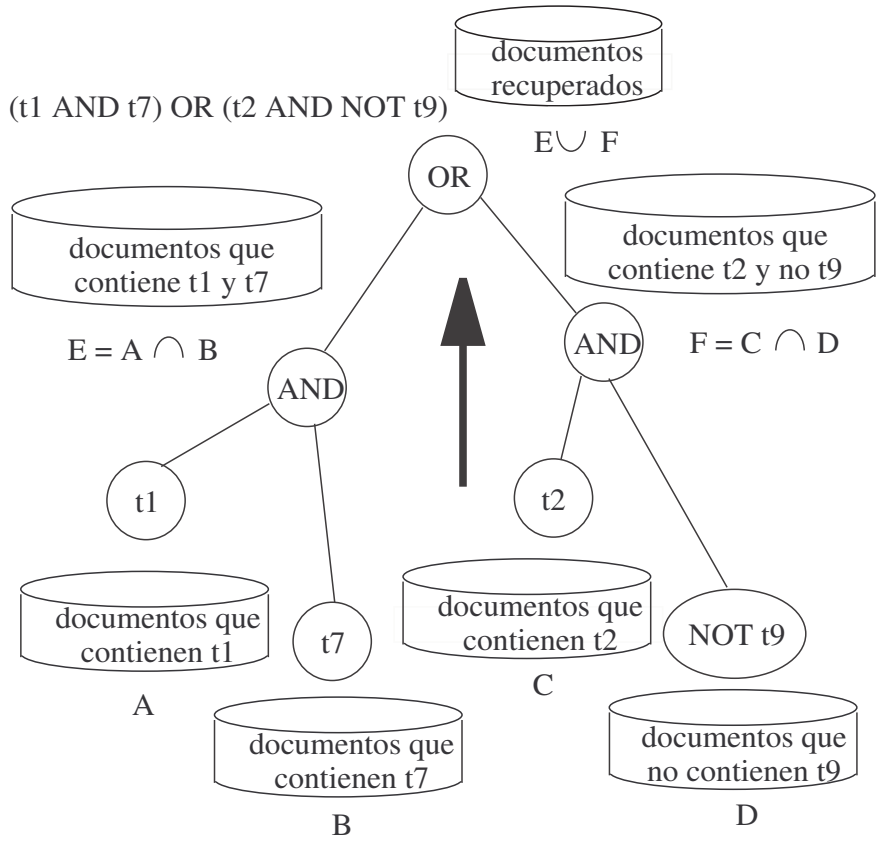

Figura 2.8: Ejemplo de evaluación en el modelo Booleano.

criterio de recuperación binario tan tajante y estricto, por lo que es más un sistema de recuperación de datos que de información.

2.3.2. Modelo Espacio Vectorial

Salton fue el primero en proponer los SRI basados en Espacio Vectorial **SRI-EV** a finales de los 60, dentro del marco del proyecto **SMART** [83]. Partiendo de que se pueden representar los documentos como vectores de términos, los documentos podrán situarse en un espacio vectorial de n dimensiones, es decir, con tantas dimensiones como elementos tenga el vector. Situado en ese espacio vectorial, cada documento cae entonces en un lugar determinado por sus coordenadas, al igual que en un espacio de tres dimensiones cada objeto queda bien ubicado si se especifican sus tres coordenadas espaciales. Se crean así grupos de documentos que quedan próximos entre sí a causa de las características de sus vectores. Estos grupos o clusters están formados, en teoría, por documentos similares, es decir, por grupos de documentos que serían relevantes para la misma clase de necesidades de información. En una base de datos documental organizada de esta manera, resulta muy rápido calcular la relevancia de un documento a una pregunta (su RSV), y siendo muy rápida también la ordenación por relevancia, ya que, de forma natural, los documentos ya están agrupados por su grado de semejanza. En la fase de la consulta, cuando se formula una pregunta, también se la deja caer en este espacio vectorial y, así, aquellos documentos que queden más próximos a ella serán, en teoría, los más relevantes para la misma. La representación de los documentos y las consultas se realiza mediante la asociación de un vector de pesos no binarios (un peso por cada término de índice). Por ejemplo, $d_i = (t_{i1}, t_{i2}, t_{i3}, \ldots, t_{in})$.

El hecho de que tanto los documentos como las consultas tengan la misma repre-

sentación dota al sistema de una gran potencialidad.

Indización de Documentos en el Modelo Vectorial

Sea $\mathcal{D}$ el conjunto de documentos y $\mathcal{T}$ el conjunto de términos índice. El mecanismo de indización de este modelo se presentará de la siguiente forma:

$$\mathcal{F} : \mathcal{D} \times \mathcal{T} \longrightarrow I$$

Lo más habitual será trabajar con una función de evaluación normalizada donde los vectores tengan los pesos reales, donde $I = [0, 1]$. Como hemos dicho anteriormente, una de las múltiples formas de definir la función $\mathcal{F}$ es la frecuencia inversa del documento (idf) [80, 83, 85]. La bondad de la indización idf está en que pondera la importancia de los términos en función de su aparición en el resto de los documentos de la base documental además de su frecuencia de aparición en el documento actual.

El Subsistema de Consulta en el Modelo Vectorial

Como hemos indicado, en este modelo tanto las consultas como los documentos tienen la misma representación, es decir, vectores n-dimensionales, donde n es el número de términos índice considerados. Cada una de las posiciones del vector contiene un peso, el cual indica la importancia relativa del término concreto de la consulta o del documento. Este peso es un número real positivo que puede estar o no normalizado. Cuando un usuario formula una pregunta, la mayoría de los pesos de la misma serán 0, con lo que bastará con proporcionar los términos con peso distinto de 0 para poder definirla. El sistema se encargará de representar la consulta completa en forma de vector n-dimensional de modo automático.

Una de las diferencias que existen entre este modelo y el Booleano es que los términos individuales considerados en la consulta no están conectados por ningún operador (ni conjunción, ni disyunción, ni negación). En el modelo vectorial, la consulta se considera como un todo. La ventaja del modelo vectorial es que permite hacer correspondencias parciales, es decir, ordena los resultados por grado de relevancia. Su principal inconveniente es que no incorpora la noción de correlación entre términos (problema de todos los modelos clásicos). Aunque este modelo se creó hace cuatro décadas y se ha investigado mucho sobre él, no se ha extendido su uso en los SRI comerciales, donde sigue demandándose el modelo Booleano a pesar de todos sus inconvenientes.

El Subsistema de Evaluación en el Modelo Vectorial

El mecanismo de evaluación de los SRI-EV empareja la consulta $\mathcal{Q}$ contra la representación (el vector) asociado a cada documento de la base documental, $d_i \in \mathcal{D}$, para obtener el grado de relevancia RSV_i del documento d_i con respecto a la consulta. El RSV toma un valor real que será tanto mayor cuanto más similares sean documento y consulta.

Existen diferentes funciones para medir la similitud entre documentos y consultas. Todas ellas están basadas en considerar ambos como puntos en un espacio n-dimensional. Como ejemplo, citaremos las siguientes:

producto escalar:

$$RSV(q,d) = \sum_{j=1}^{n} d_j \cdot q_j$$

donde d_j y q_j son, respectivamente, los pesos asociados al término t_j en la representación del documento d y la consulta q.

medida del coseno:

$$RSV(q,d) = \frac{\sum_{j=1}^{n} d_j \cdot q_j}{\sqrt{\sum_{j=1}^{n} d_j^2 \cdot q_j^2}}$$

índice de Dice:

$$RSV(q,d) = \frac{2 \cdot \sum_{j=1}^{n} d_j \cdot q_j}{\sum_{j=1}^{n} (d_j^2 + q_j^2)}$$

índice de Jaccard:

$$RSV(q,d) = \frac{\sum_{j=1}^{n} d_j \cdot q_j}{\sum_{j=1}^{n} (d_j^2 + q_j^2 - d_j \cdot q_j)}$$

distancia euclídea: Calcula la distancia existente entre ambos vectores en el espacio:

$$RSV(q,d) = -\sqrt{\sum_{j=1}^{n} d_j^2 - q_j^2}$$

2.3.3. Modelo Probabilístico

El marco del modelo probabilístico está compuesto por conjuntos de variables, operaciones con probabilidades y el teorema de Bayes.

Todos los modelos de recuperación probabilísticos están basados en el que hemos traducido como el *Principio de la ordenación por probabilidad*, conocido originalmente como "the probability ranking principle". Este principio, formulado por Robertson en [78], asegura que el rendimiento óptimo de la recuperación se consigue ordenando los documentos según sus probabilidades de ser juzgados relevantes con respecto a una consulta, siendo estas probabilidades calculadas de la forma más precisa posible a partir de la información disponible. Así, y atendiendo a este principio, el objetivo primordial de cualquier modelo probabilístico, pasa por calcular $p(R|qd_i)$.

Comencemos esta revisión de los modelos probabilísticos por el primero que surgió, el conocido como *modelo de recuperación con independencia binaria*, en inglés "Binary

Independence Retrieval (BIR)", que fue inicialmente planteado por Maron y Kuhns en [67], continuado por Robertson y Spark Jones [79] y concluido por van Rijsbergen en [90].

En él, los documentos y las consultas se representan por un vector binario. Así, un documento cualquiera tiene la siguiente forma:

$$d_j = (t_1, t_2, \ldots, t_n)$$

donde $t_i = 0$ ó 1 indica la ausencia o presencia del término i-ésimo, respectivamente, y n el número de términos de la colección. Existen dos eventos mutuamente excluyentes: ω_1, que representa el hecho de que un documento sea relevante, y ω_2, que indica que no lo sea. Este modelo asume que se conocen, o por lo menos se suponen, el conjunto de documentos relevantes (R) y no relevantes ($\overline{R}$) de una consulta dada.

El objetivo que se persigue es calcular $p(\omega_1|d_j)$ y $p(\omega_2|d_j)$, decir, la probabilidad de que el documento d_j sea relevante y no relevante, respectivamente, dada una consulta q y desarrollar una función que ofrezca un valor de relevancia para así poder ordenar los documentos según ella. En este caso, esa función tendrá la forma:

$$Sim(d_j, q) = \frac{p(\omega_1|d_j)}{p(\omega_2|d_j)}. \tag{2.2}$$

Haciendo suposiciones de independencia entre términos y aplicando el teorema de Bayes, se llega a:

$$Sim(d_j, q) \sim \sum_{i=1}^{n} log(\frac{p(t_i = 1|\omega_1) \cdot (1 - p(t_i = 1|\omega_2))}{p(t_i = 1|\omega_2) \cdot (1 - p(t_i = 1|\omega_1))})t_i + c, \tag{2.3}$$

donde

	Relevante	No Relevante	
Aparece	n_i^R	$n_i - n_i^R$	n_i
No aparece	$\|R\| - n_i^R$	$N - n_i - \|R\| + n_i^R$	$N - n_i$
	$\|R\|$	$N - \|R\|$	N

Tabla 2.1: Distribución de la aparición o no de un término en los documentos relevantes y no relevantes.

$$c = \sum_{i=1}^{n} log(\frac{1 - p(t_i = 1|\omega_1)}{1 - p(t_i = 1|\omega_2)}), \tag{2.4}$$

siendo $p(t_i = 1|\omega_1)$ la probabilidad de que un término t_i esté presente en el conjunto de documentos relevantes y $p(t_i = 1|\omega_2)$ en los no relevantes. El logaritmo que multiplica al peso binario t_i, en la expresión 2.3 se conoce como el *peso de relevancia del término*: el valor que se le asigna a cada término cuando se está llevando a cabo una indexación probabilística, expresando la capacidad de discriminación de éste entre documentos relevante y no relevantes.

La Tabla 2.1 representa una tabla de contingencia para un término de la colección y muestra la distribución de apariciones o no del término i-ésimo en los documentos relevantes y no relevantes para una consulta. Dado que R es el conjunto de documentos relevantes, y $|R|$ su cardinal, N es el número total de documentos de la colección, n_i es el número de documentos en los que aparece t_i y n_i^R es el número de veces que aparece el término en documentos relevantes, las probabilidades $p(t_i - 1|\omega_1)$ y $p(t_i - 1|\omega_2)$ se estiman según las siguientes expresiones:

$$p(t_i = 1|\omega_1) = \frac{n_i^R}{|R|}; p(t_i = 1|\omega_2) = \frac{N - n_i^R}{N - |R|} \tag{2.5}$$

El uso del modelo probabilístico que se acaba de presentar es el siguiente: el usuario formula una consulta al SRI y éste, mediante la expresión 2.3, calcula un valor de

relevancia para cada documento, generando así una lista ordenada de documentos. Cuando el usuario ha formulado una primera consulta, el SRI no tiene información para poder estimar $p(t_i = 1|\omega_1)$ y $p(t_i = 1|\omega_2)$, según las expresiones 2.5, por lo que se deben establecer estimaciones iniciales, a partir de la colección completa, que pueden ser [4]:

$$p(t_i = 1|\omega_1) = 0.5; p(t_i = 1|\omega_2) = \frac{n_i}{N}. \tag{2.6}$$

Croft y Harper ofrecen, en [29], varias estimaciones iniciales para cuando no hay información relevante y los rendimientos alcalzados con cada una de ellas. Por otro lado, Spark Jones, en [57], establece varias expresiones cuando la información de la que se dispone es muy poca para obtener las tablas de contingencia de cada término.

A partir de la primera lista de documentos, el usuario emite sus juicios de relevancia con respecto a los documentos que figuran en ella y el SRI genera la Tabla 2.1, donde sí podrá aplicar directamente las expresiones 2.6 y reiterar este proceso hasta que el usuario quede satisfecho.

Existen otros modelos probabilísticos que surgieron como variación o mejora de este anterior. Entre ellos podemos destacar el conocido como *modelo de indexación de independencia binaria* [39], que se desarrolló a partir del modelo de Maron y Kuhns. Mientras el modelo de recuperación de independencia binaria trabaja con los documentos de la colección y una consulta, este modelo trabaja con un conjunto de consultas y el peso de cada término lo calcula con respecto a las consultas que usan ese término.

2.3.4. Modelo Booleano Extendido

Cualquier SRI debe ser capaz de tratar con dos características inherentes al proceso de RI: la imprecisión y la subjetividad [13]. Estos dos factores juegan un papel fundamental en los diferentes estados de procesamiento de la información, tales como:

- en la formulación de las necesidades de información,
- en la estimación del grado en que cada ítem de información es relevante para las necesidades del usuario, y
- en la decisión de qué ítems de información deben recuperarse en función a una petición determinada.

Los SRI Booleanos no incorporan herramientas adecuadas para manejar las dos características anteriores (imprecisión y subjetividad). Debido a ello, los SRI basados en este modelo de recuperación presentan los siguientes problemas:

- Una de sus mayores inconvenientes es la indización de los documentos. Un término puede aparecer en un documento y ser más significativo en éste que en cualquier otro. Sin embargo, no existen mecanismos para representar esta distinción en el modelo Booleano. Este inconveniente afecta directamente al módulo indizador de la base documental.
- Otra fuente de imprecisión que caracteriza a la RI es el conocimiento vago que el usuario tiene sobre el tema sobre el que está preguntando. Si el usuario es un entendido, le gustaría tener la habilidad de expresar en su consulta la importancia o relevancia que tienen unos términos sobre otros, es decir, expresar la importancia relativa a través del lenguaje de consulta. La incapacidad de realizar esta tarea

viene a ser una carencia muy representativa del subsistema de consulta de los SRI Booleanos.

- Por último, la recuperación será tajante: 1 si el documento es relevante y 0 si no lo es. El RSV será 0 o 1, sin permitir que exista una gradación en la recuperación que maneje mejor la incertidumbre. Este problema se centra en el mecanismo de evaluación.

Sin embargo, a pesar de las carencias anteriores, el modelo Booleano sigue estando muy extendido en el ámbito comercial. Por esta razón, se han llevado a cabo varias extensiones sobre el mismo que permiten salvar algunas de las limitaciones que presenta sin proceder a su completa redefinición. La teoría de conjuntos difusos [98] se ha empleado como herramienta para tal propósito, especialmente por su habilidad para tratar con la imprecisión y la incertidumbre en el proceso de RI. Este hecho se debe fundamentalmente a dos razones principales [12]:

- es un marco formal diseñado para tratar con imprecisión y vaguedad, y
- facilita la definición de una superestructura del modelo Booleano, de forma que los SRI basados en este modelo pueden modificarse sin tener que ser completamente rediseñados.

El modelo Booleano extendido (SRI-BE), resultante de la aplicación de las técnicas difusas al modelo Booleano, extiende a este último en tres aspectos principales.

Indización en el Modelo Booleano Extendido

En primer lugar, la indización de los términos se llevará a cabo del mismo modo que en el modelo Espacio Vectorial, que permite que un documento tenga asociado un peso para cada término, que indica el grado en que el documento se caracteriza por tal

término. Los pesos toman valor en el rango [0,1]. Se basará por tanto en una indización difusa donde una función de pertenencia $\mathcal{F}$ mostrará el grado en el que el término representa al documento.

Dentro del marco difuso, los documentos se representarán como conjuntos difusos de términos índice en los cuales el grado de pertenencia, que liga un término a un documento, expresa si el término describe el contenido del docuemento de manera significativa.

Por tanto, esta consideración se podría interpretar como una función de pertenencia de un conjunto bidimensional [58, 101] (una relación difusa) que muestra el grado en que el documento d pertenece a ese grupo de documentos que pertenecen al/los concepto/s representado/s por un término t. De tal forma, se podría asociar un conjunto difuso a cada documento y término como sigue:

$$d_i = \{\langle t, \mu_{d_i}(t)\rangle | t \in \mathcal{T}; \mu_{d_i}(t) = \mathcal{F}(d_i, t)\}$$

$$t_j = \{\langle d, \mu_{t_i}(d)\rangle | d \in \mathcal{D}; \mu_{t_i}(t) = \mathcal{F}(d, t_j)\}.$$

El Subsistema de Consulta en el Modelo Booleano Extendido

Al igual que en el modelo Espacio Vectorial, el RSV de los documentos será un valor gradual, que en este caso estará en el intervalo [0,1]. Esto permite la aparición de una relevancia parcial y permite ordenar los resultados en función a su valor.

El conjunto final de documentos recuperados puede venir definido por dos vías distintas: bien proporcionando un umbral superior para el número de documento recu-

perados o bien definiendo un umbral α para el grado de relevancia (esta última opción conlleva obtener el α-corte del conjunto difuso resultante de la consulta $\mathcal{Q}$).

Por tanto, considerando de ese modo, el conjunto final de documentos recuperados sería:

$$R = \{d \in \mathcal{D} | RSV_q(\mathcal{D}) \geq \alpha\}$$

Por otro lado, también se produce una extensión en el lenguaje de consulta Booleano. Dentro del marco actual, se introducen factores de peso numéricos, que pueden afectar tanto a los términos como a los operadores Booleanos. Incluso, recientemente, varios autores han propuesto extensiones basadas en el uso de términos lingüísticos en lugar de pesos numéricos, lo que facilita la labor de difinición de la consulta al usuario [11, 51].

Así, esta extensión del lenguaje de consulta Booleano utilizando la teoría de conjuntos difusos enfoca ahora el problema en componer criterios de selección más expresivos utilizando pesos numéricos en las consultas.

Un ejemplo de consulta Booleana extendida sería:

$$(\langle w_7, t_7 \rangle OR \langle w_2, t_2 \rangle) AND (\langle w_1, t_1 \rangle AND NOT \langle w_5, t_5 \rangle)$$

donde w_1, w_2, w_5, w_7 son pesos numéricos definidos en [0,1] (o términos lingüísticos con un conjunto difuso que define su semántica en el modelo lingüístico).

Como veremos a continuación, estos pesos se definen con diferentes semánticas para permitir al usuario cuantificar la importancia de los criterios de selección. La semántica

considerada afectará al funcionamiento del mecanismo de evaluación y, en consecuencia, al RSV de los documentos recuperados.

El Subsistema de Evaluación en el Modelo Booleano Extendido

De este modo, la diferencia principal entre el subsistema de consulta del modelo Booleano y el del modelo Booleano extendido es la aparición de pesos y el hecho de que el resultado de la consulta sea un conjunto difuso definido sobre el espacio de los documentos. Este concepto de consultas ha generado el problema de la interpretación de los pesos.

El proceso de evaluación de la consulta se realiza desde abajo hacia arriba, empezando por los términos simples de la consulta. El primer paso consiste en combinar cada término individual con su peso asociado, obteniendo el RSV de cada documento para la consulta compuesta por un único término y su peso. Esta operación se realiza mediente el operador $\mathcal{E}(d, \langle t, w \rangle)$, cuya definición depende de la interpretación asociada a los pesos como veremos a continuación. Posteriormente, se pasa a calcular el valor de la recuperación final como resultado de las combinaciones Booleanas de las $\mathcal{E}(d, \langle t, w \rangle)$ parciales.

El operador difuso asociado a los operadores Booleanos es el mismo, independientemente de la interpretación de los pesos. En principio, el operador **AND** se interpreta como el mínimo, el **OR** como el máximo y el **NOT** como la función 1-x (aunque es posible utilizar otros operadores difusos t-norma, t-conorma y función de negación) [58, 101]. Este mecanismo de evaluación garantiza el *principio de separabilidad* de la lista de peticiones que es satisfactorio en todos los casos salvo en uno, cuando los pesos

se interpretan con la semántica de importancia relativa como se verá en los Capítulos 4 y 5.

Diferentes autores han reconocido que las semánticas de los pesos en la consulta deberían estar relacionadas con el concepto de *importancia* del término, pero la duda es que cómo pueden las consultas Booleanas ponderadas representar la generación de las Booleanas simples y saber cuál es al relación semántica entre los pesos de los términos índice.

En respuesta a estas dudas, se han introducido diferentes semánticas para los grandos de pertenencia asociados con el término t en la definición de la consulta, tales como:

- la *importancia relativa* de t, que permite al usuario expresar la importancia de cada término en la consulta [8, 77, 84],
- el *umbral* para t, que considera los pesos como umbrales, premiando al documento cuyo grado de pertenencia para el término t sea mayor o igual que el grado de pertenencia del término en la consulta pero permitiendo algún valor de coincidencia parcial cuando el grado de pertenencia del documento es menor que el umbral [20, 77],
- el *documento perfecto* para el término t con respecto a la evaluación del documento [10, 21], que especifica que la descripción difusa de la consulta representa qué descripción ideal difusa del documento debería darse para satisfacerla. Las semánticas de la perfección deben ser referidas únicamente como importancia absoluta.

Como ya hemos comentado, una de las ventajas de aplicar estas extensiones a los SRI Booleanos es que los documentos podrán ser ordenados según el grado de pertenencia, es decir, en función de su relevancia. El usuario podrá limitar el número de documentos recuperados.

Consideremos consultas en las que únicamente se ponderan los términos y no los operadores, la función de evaluación global $\mathcal{E} : \mathcal{D} \times \mathcal{Q} \rightarrow [0,1]$ está definida sobre la colección de documentos $\mathcal{D}$ y sobre el conjunto de consultas legítimas $\mathcal{Q}$ obtenidas mediante la aplicación de reglas sintácticas siguientes:

1. $\forall \langle t, w \rangle \in \mathcal{T} \times [0,1] \Rightarrow \langle t, w \rangle \in \mathcal{Q}$
2. $\forall q, p \in \mathcal{Q} \Rightarrow qANDp \in \mathcal{Q}$
3. $\forall q, p \in \mathcal{Q} \Rightarrow qORp \in \mathcal{Q}$
4. $\forall q \in \mathcal{Q} \Rightarrow NOTq \in \mathcal{Q}$
5. Sólo se pueden obtener consultas Booleanas extendidas aplicando las reglas 1-4.

En vista de las anteriores reglas de ampliación y asumiendo la definición normalizada de $\cap$, $\cup$ y $\neg$ para conjuntos difusos como el mínimo, el máximo y el complemento, respectivamente tenemos:

$$\mathcal{E}(q_1 AND q_2) = \mathcal{E}(q_1) \cap \mathcal{E}(q_2)$$

$$\mathcal{E}(q_1 OR q_2) = \mathcal{E}(q_1) \cup \mathcal{E}(q_2)$$

$$\mathcal{E}(NOTq) = \neg \mathcal{E}(q)$$

donde q, q_1, $q_2 \in \mathcal{Q}$.

2.4. Evaluación de los Sistemas de Recuperación de Información

Un SRI puede evaluarse empleando diversos criterios. Frakes [38] selecciona los dos siguientes como los más importantes: ejecución eficaz (eficacia). La importancia relativa de estos factores debe decidirla el diseñador del sistema, y la selección de la estructura de datos y los algoritmos apropiados para su implementación dependerá de esa decisión.

La eficacia en la ejecución se medirá por el tiempo que toma el sistema o una parte del mismo para llevar a cabo una operación. Este parámetro ha sido siempre una preocupación principal en un SRI, especialmente desde que muchos de ellos son interactivos y un tiempo de recuperación excesivo interfiere con la utilidad del sistema, llegando a alejar a los usuarios del mismo. Los requerimientos no funcionales de un SRI normalmente especifican el tiempo máximo aceptable para una búsqueda y para las operaciones de mantenimiento de una base documental, tales como añadir y borrar documentos.

La eficiencia del almacenamiento se medirá por el número de bytes que se precisan para almacenar los datos. El espacio general, una forma común de medir la eficacia del almacenamiento, es la razón del tamaño de los ficheros índice más el tamaño de los archivos del documento sobre el tamaño de los archivos del documento.

Tradicionalmente, se le ha dado mucha importancia a la efectividad de la recuperación, normalmente basada en la relevancia de los documentos recuperados a las necesidades reales de información del usuario, lo cual ha representado un problema ya que medir la relevancia es un proceso subjetivo y sin confianza. Esto es, diferentes juicios

personales asignarían diferentes valores de relevancia a un documento recuperado en respuesta a una búsqueda.

Por otro lado, Salton y McGill [83] señalan que, además de los criterios anteriores que se centran principalmente en el punto de vista del diseñador del sistema, se debe considerar también el punto de vista del usuario ya que los criterios de evaluación del diseñador y del usuario no tienen por qué coincidir. Los seis criterios siguientes han sido identificados como los más importantes en lo que respecta a las características que un SRI debe ofrecer al usuario [25, 63]:

1. La exhaustividad, o habilidad del sistema para presentar todos los items relevantes.

2. La precisión, o habilidad del sistema para presentar solamente items relevantes.

3. El esfuerzo, intelectual o físico, requerido por el usuario en la formulación de las consultas, en el manejo de la búsqueda y en el proceso de examinar los resultados.

4. El intervalo de tiempo transcurrido entre que el sistema recibe la consulta del usuario y presenta las respuestas.

5. La forma de presentación de los resultados de la búsqueda, la cual influye en la habilidad del usuario para utilizar la información recuperada.

6. El alcance o cobertura de la colección documental, o la proporción en la que están incluidos en la recuperación todos los items relevantes del sistema ya conocidos por el usuario.

Una vez repasados los distintos factores que se pueden considerar en el proceso de evaluación de un SRI, en importante destacar el hecho de que el propio concepto de

evaluación puede verse desde dos perspectivas distintas en el área [4, 38, 80, 83, 90]. Existen dos grandes corrientes de investigación en evaluación de la RI, denominadas respectivamente la corriente algorítmica, basada en el modelo tradicional de evaluación, y la corriente cognitiva. Mientras que el primer modelo, el más antiguo, se centra en los algoritmos y en las estructuras de datos necesarias para optimizar la eficacia y la eficiencia de las búsquedas en bases documentales, el segundo, más reciente, considera el papel del usuario y de las fuentes de conocimiento implicadas en la RI [56].

La base del modelo cognitivo la constituyen los trabajos de Dervin y Nilan [33], y Ellis [34], que buscaban proponer una alternativa al modelo clásico de evaluación. Este modelo ha provocado un interés creciente por incorporar a los usuarios en el proceso de evaluación, considerando ésta desde el punto de vista del propio usuario final del SRI.

Desde esta perspectiva, la evaluación se centra en la representación de los problemas de información, el comportamiento en las búsquedas y los componentes humanos de los SRI en situaciones reales, y se fundamenta en la psicología cognitiva y en las ciencias sociales. La búsqueda de información y la formulación de la necesidad de información se contemplan como procesos cognitivos del usuario individual, siendo el SRI y los intermediarios funcionales (como la interfaz del sistema) componentes fundamentales de este proceso de contextualización [72].

La naturaleza compleja de las necesidades de información han puesto de manifiesto que la investigación orientada solamente a técnicas algorítmicas de RI no puede ofrecer una panorámica global del proceso de recuperación. Para lograrla, es necesario incorporar las características del sistema, las características situacionales del usuario y los

intermediarios imprescindibles, el más importante de los cuales es la interfaz de usuario, al ser el mecanismo principal de enlace entre este último y el sistema [56].

En el marco del modelo cognitivo, se han propuesto distintas medidas de evaluación del SRI relacionadas con el concepto de relevancia basada en el usuario. Entre ellas se encuentran la proporción de cobertura o alcance ("coverage ratio"), definida como la fracción de documentos relevantes conocidos por el usuario que han sido recuperados, y la proporción de novedad ("novelty ratio"), que se define como la fracción de documentos relevantes recuperados que son desconocidos por el usuario [4, 60]. También se ha considerado la satisfacción del usuario como medida de la eficacia del SRI en este marco de trabajo [60], aunque no se ha propuesto una forma adecuada para medirla al ser un criterio muy subjetivo.

Precisamente, esta última es la mayor crítica al modelo cognitivo, que también utiliza otras medidas como beneficios y frustraciones, utilidad, etc. que no son objetivas y que no evalúan directamente el sistema sino el efecto que provoca en el usuario.

Para finalizar esta sección, retomaremos el modelo algorítmico de evaluación de la RI para describir algunas de las medidas de eficacia habitualmente consideradas en él, puesto que nos referiremos a ellas en el resto de la memoria.

Se han propuesto múltiples medidas de efectividad de la RI, siendo las más empleadas y conocidas la exhaustividad y la precisión [90].

La exhaustividad, E, es la proporción de documentos relevantes recuperados en una

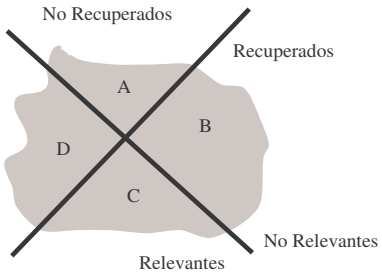

Figura 2.9: Distribución de documentos en el proceso de recuperación.

búsqueda determinada sobre el número de documentos relevantes para esa búsqueda en la base de datos, siendo su fórmula:

$$E = \frac{C}{B + C}$$

La precisión es la proporción de documentos relevantes recuperados sobre el número total de documentos recuperados, siendo su fórmula:

$$P = \frac{C}{D + C}$$

donde C = número de documentos relevantes recuperados, B = número de documentos relevantes no recuperados y D = número de documentos no relevantes recuperados.

En la Figura 2.9 se puede ver de manera gráfica la distribución de estos conjuntos de documentos en el proceso de recuperación.

En tanto que se quiere comparar la efectividad del SRI en los términos de exhaustividad y precisión, se han desarrollado métodos para evaluarlos de forma simultánea. De este modo, es habitual trabajar con un sistema de coordenadas en el que un eje es

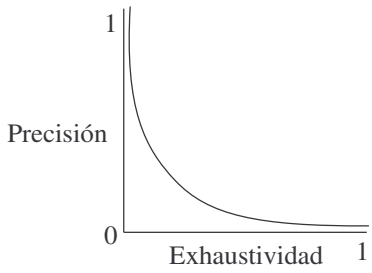

Figura 2.10: Precisión vs exhaustividad.

para exhaustividad y otro para precisión (véase la Figura 2.10).

En teoría, los puntos de exhaustividad-precisión están inversamente relacionados [24]. Esto es, cuando la precisión sube, la exhaustividad normalmente baja y viceversa. Esto se debe a que, mientras que la precisión da importancia a la ausencia o "no recuperación" de documentos no relevantes, la exhaustividad se basa fundamentalmente en la recuperación de todos los documentos relevantes, aunque esto implique recuperación de documentos no relevantes. Por tanto, el fin de la precisión es la ausencia de lo que en el ámbito documental llamamos "ruido", mientras que en la exhaustividad lo que se intenta evitar es el "silencio".

Una medida de evaluación combinada de exhaustividad y precisión, M, fue desarrollada por Van Rijsbergen [90] y definida como:

$$M = 1 - \frac{(1 + b^2)P \cdot E}{b^2 \cdot P + E}$$

donde P = precisión, E = exhaustividad, y b es una medida de la importancia relativa, para un usuario, de exhaustividad y precisión. Los investigadores suelen manejar valores de M que reflejen la exhaustividad y precisión que interese al usuario típico.

Como todos los SRI, los basados en el modelo vectorial (SRI-EV) y en el modelo booleano extendido (SRI-BE) también se evalúan considerando las medidas de precisión y exhaustividad. Sin embargo, debido a que los SRI-EV y SRI-BE devuelven todos los documentos de la base como respuesta a una consulta concreta, es necesario aplicar un tratamiento especial.

Una posibilidad consiste en trabajar con una filosofía de umbral (α-corte para SRI-BE) y considerar como conjunto de documentos recuperados final aquel que se obtiene una vez aplicado dicho umbral. En ese caso, se trabajaría de la misma manera que en el caso de los SRI Booleanos.

Sin embargo, este modo de trabajo presenta varios inconvenientes:

- No es bueno que exista una dependencia del umbral en la evaluación del sistema. Una mala elección de éste puede llevar a pensar que el SRI responde mal a la consulta cuando en realidad lo está haciendo correctamente.
- Puesto que los SRI-EV y SRI-BE presentan la potencialidad de devolver los documentos ordenados según su relevancia, sería interesante que el mecanismo de evaluación fuese capaz de medir también esta capacidad.

Por estas razones, lo más habitual consiste en ignorar el umbral y estudiar el comportamiento del sistema ante una consulta considerando todos los documentos recuperados, es decir, todos los de la base junto con su RSV asociado. Para ello, se procede del siguiente modo:

1. Se fija un conjunto creciente de p valores de exhaustividad equidistante en [0,1]. Se

consideran estos valores como marcas en el conjunto de documentos recuperados. Nos situamos en el primer documento de la lista.

2. Comprobamos la relevancia del documento actual. Si no es relevante, descendemos en la lista hasta localizar el siguiente documento relevante.

3. Consideramos el conjunto de documentos existentes entre el principio de la lista y el documento actual y medimos la precisión y exhaustividad obtenidas.

4. Descendemos al siguiente documento de la lista y repetimos los pasos 2 y 3 hasta procesar el último documento relevante.

5. Interpolamos los valores de precisión y exhaustividad obtenidos para calcular los asociados a los p valores fijados inicialmente.

6. Finalmente, calculamos la media de los p valores y consideramos esa medida como índice de la calidad del SRI-EV.

Observamos que un valor alto de esta medida indicará un buen comportamiento del sistema. Cuanto mayor sea este valor, más documentos relevantes habremos encontrado entre los primeros de la lista. El valor óptimo es 1 y se obtiene cuando todos los documentos relevantes se encuentran en las primeras posiciones de la lista y no hay ningún documento no relevante entre ellos. La aparición de documentos no relevantes en esas posiciones provoca que las marcas se desplacen hacia abajo y que, consecuentemente, los valores de precisión desciendan.

También es destacable que se puedan calcular tanto la precisión media para una serie de valores de exhaustividad (caso del algoritmo anterior) como la exhaustividad media para una serie de valores de precisión.

Finalmente, diremos que en cualquiera de los dos casos, son prácticas habituales escoger los once ($p = 11$) valores siguientes $\{0, 0.1, 0.2, 0.3, \ldots, 0.9, 1\}$, o los tres ($p = 3$) siguientes $\{0.25, 0.5, 0.75\}$.

2.5. Métodos para Mejorar la Recuperación de Información

Hay veces en que el usuario no es capaz de encontrar una consulta que exprese fielmente su necesidad de información u ocasiones en las que, por limitaciones del propio SRI, éste no consigue recuperar todos los documentos relevantes. Por estas y otras razones se han desarrollado técnicas que permiten asistir al usuario a la hora de formular la consulta, por un lado, y por otro, reformular la consulta de manera iterativa a la luz de los juicios de relevancia expresados por el usuario. Algunas de estas técnicas son:

Tesauros. En primer lugar, y en cuanto a que ayuda al usuario para formular la consulta, destacamos el uso de los tesauros: un conjunto de palabras y las relaciones que existen entre sí, las cuales van desde sinonimias y antonimias hasta cualquier otro tipo de relación entre ellas. El tesauro puede usarlo el propio usuario para expresar su necesidad de información, mediante la búsqueda de las palabras adecuadas o, alternativamente, se suele utilizar como fuente para añadir nuevos términos a una consulta, proceso que se conoce como expansión de consultas.

Realimentación de relevancia. Por otro lado, siguiendo el estilo de trabajo que tiene el modelo probabilístico, existe la técnica conocida como realimentación de relevancia, la cual parte de un conjunto de juicios de relevancia expresados por el usuario tras una primera recuperación. Con esa información suministrada al SRI, éste modifica la consulta de dos maneras:

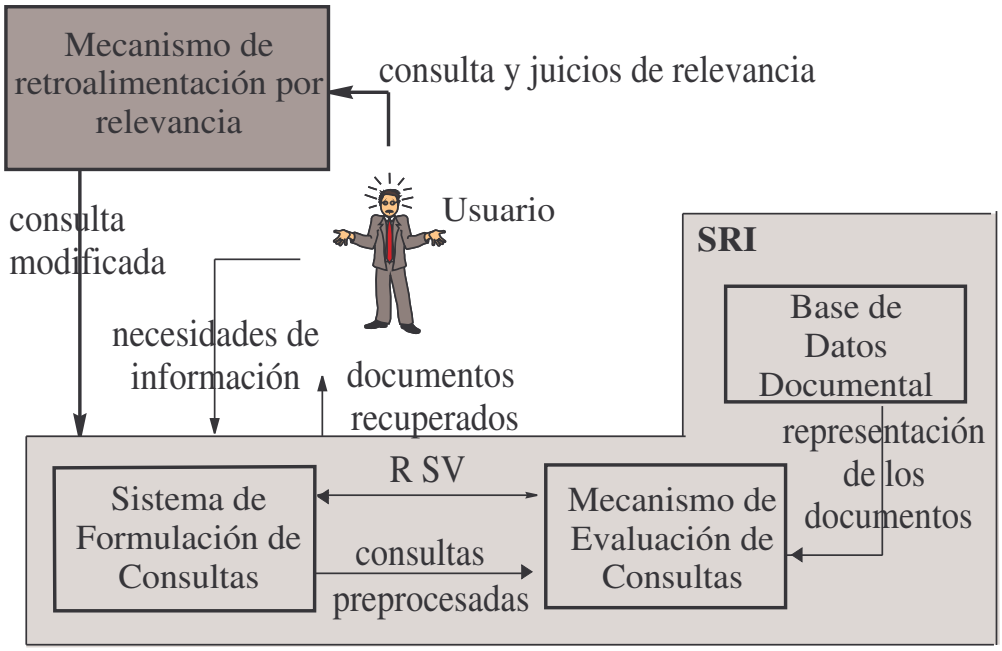

Figura 2.11: Proceso de retroalimentación por relevancia.

- Alterando los pesos de los términos que componen la consulta original, de tal forma que se le da más fuerza a aquéllos que aparecen más en documentos relevantes que en no relevantes, y debilitando a los que ocurren en la situación contraria. Esta técnica se conoce como repesado de los términos.
- Añadiendo nuevos términos que no aparecen en la consulta original, pero que sí lo hacen en los documentos que han sido juzgados como relevantes o no relevantes. Este proceso es también conocido como expansión de consultas.

Aprendizaje automático de consultas. Uno de los problemas principales que los usuarios no expertos encuentran cuando se enfrentan a un SRI es la necesidad de conocer en profundidad el lenguaje de consulta del mismo para poder expresar sus necesidades de información en forma de una consulta interpretable por el sistema, que les permita recuperar información relevante. Para resolver el problema

de la formulación de consultas en diferentes clases de SRI, se han desarrollado diferentes aproximaciones para ayudar al usuario en dicho proceso [23]. Una de las más conocidas se basa en la generación automática de consultas que describan adecuadamente las necesidades del usuario -(representadas en forma de un conjunto inicial de documentos relevantes (y opcionalmente no relevantes)- mediante un proceso *off-line* en el que su interacción no es necesaria. Esta operación se incluye en el paradigma de Aprendizaje Automático [69] y que Chen y otros la han denominado Aprendizaje Inductivo de Consultas a partir de Ejemplos (IQBE) [23].

La consulta obtenida de este proceso podrá ejecutarse en otros SRI para obtener nuevos documentos relevantes. De esta forma, tal y como se muestra en la Figura 2.12, no es necesario que el usuario interactúe con el sistema como en otras técnicas de refinamiento de consultas como la retroalimentación por relevancia (véase Figura 2.11), analizada en la sección anterior.

Es importante señalar que, en la mayoría de los casos, la diferencia entre el IQBE y la retroalimentación por relevancia es muy sutíl. Habitualmente, dicha diferencia se centra únicamente en la existencia o no de la interacción del usuario, la cual es consecuencia del objetivo final del proceso. En el caso de la retroalimentación, esta interacción está presente al encontrarnos en un proceso cíclico, en línea, que persigue refinar (iterativamente y en varios pasos) una consulta ya existente para obtener nuevos documentos relevantes. En cambio, el objetivo final del IQBE es asistir al usuario en el proceso de formulación de la consulta derivando automáticamente una consulta que represente un conjunto de documentos relevantes

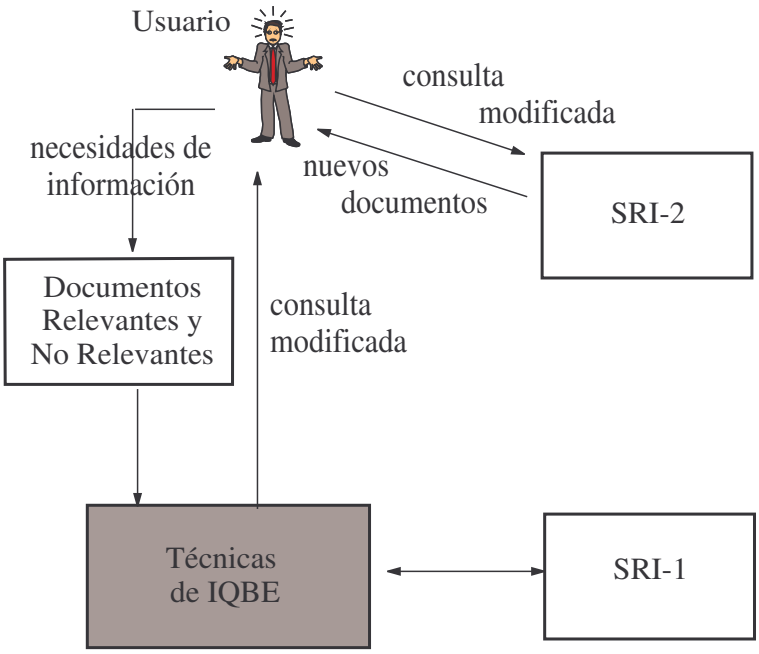

Figura 2.12: Proceso de Inductive Query by Example.

e irrelevantes para sus necesidades de información, por lo que es un proceso fuera de línea, en un único paso y en el que no se requiere ninguna acción por su parte (aparte, lógicamente, de la tarea previa de determinar el conjunto de documentos inicial).

2.6. Filtrado de Información versus Recuperación de Información

Hoy en día el acceso a la información en Internet es una actividad compleja para la que los usuarios necesitan sistemas que los ayuden en la selección de la información que les interesa. Los diferentes sistemas propuestos hacen uso de métodos, conceptos

y técnicas procedentes de diversas áreas de investigación, tales como: recuperación de información, filtrado de información, inteligencia artificial o ciencia del comportamiento.

A pesar de estar basados en diferentes filosofías, todos estos sistemas comparten su objetivo principal, exponer a los usuarios solamente la información relevante para ellos, de forma que empleen de manera optima el tiempo con que cuentan [41].

De hecho, Belkin y Croft [5] determinaron que el filtrado de información (FI) y la RI son las dos caras de una misma moneda, que trabajan conjuntamente para ayudar a los usuarios en la obtención de información que necesitan para lograr sus objetivos. Usando sistemas de filtrado de información (SFI) podemos depurar la información seleccionada por los SRI de forma que la que sea mostrada a los usuario se adapte lo más posible a sus necesidades.

Así, el FI es un proceso de búsqueda de información donde las necesidades de información del usuario perduran en el tiempo [41, 71] y cuya idea es muy simple. Un usuario proporciona sus necesidades de información a un SFI y éste le devuelve una serie de documentos de forma que él indicará que documentos recuperados son relevantes para sus necesidades y cuales no. Una vez obtenida esta información, se procede a almacenar los documentos relevantes e irrelevantes para esa necesidad de información específica y para ese usuario (perfil de usuario). De esta forma, cuando un usuario vuelve a realizar una consulta con unas necesidades de información similares, el sistema tendrá en cuenta qué documentos marcó el usuario como irrelevantes y los tratará como tales. En la Figura 2.13 se representa el funcionamiento de este tipo de sistemas.

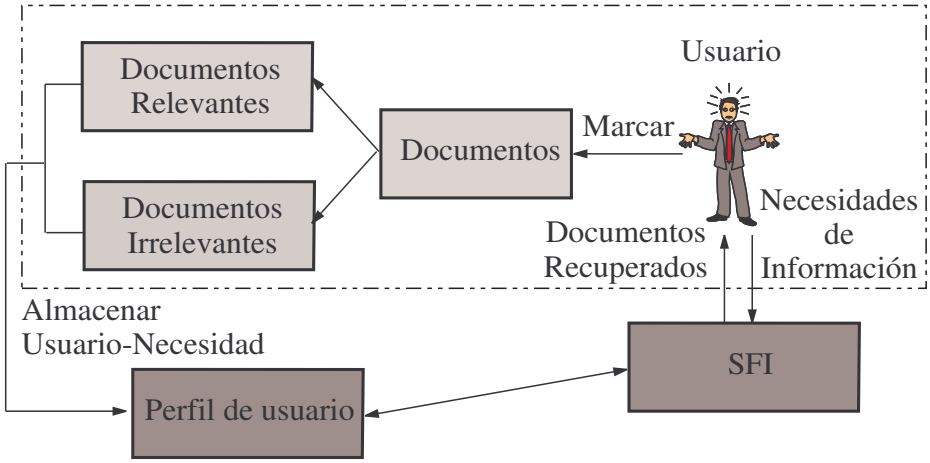

Figura 2.13: Perfil de usuario.

A pesar de que ambos sistemas tienen en común su objetivo, la selección de información relevante, difieren en otra serie de aspectos. La Tabla 2.2 recoge algunas de las diferencias existentes.

Las necesidades de información de un usuario en un SFI se representan por medio de un "perfil". La estructura más común de perfil es la denominada "bag of words", la cual consiste en un conjunto de palabras clave que representan los intereses del usuario.

Muchos de los sistemas que trabajan con perfiles asumen que será el propio usuario el que defina el perfil, identificando las palabras que lo formaran. Sin embargo, esta forma de trabajo lleva ligada la dificultad con la que se encuentra el usuario a la hora de seleccionar las palabras adecuadas para comunicarse con el sistema. Es lo que clásicamente se conoce como el "problema del vocabulario" en la interacción humano-ordenador [40].

Parámetro	Recuperación de Información	Filtrado de Información
Frecuencia de Uso	Uso puntual	Uso repetitivo. Usuarios con necesidades constantes de información
Representación de las necesidades de información	Consultas	Perfiles
Bases de datos	Relativamente estáticas	Datos dinámicos
Tipo de usuarios	El sistema no tiene conocimiento sobre ellos	El sistema tiene almacenados los par perfiles de los usuarios

Tabla 2.2: Comparación entre RI y FI.

Debido a esta razón, se han aplicado técnicas de aprendizaje automático a la construcción implícita de perfiles [35, 71]. En estos casos, el perfil se aprende automáticamente a partir de un conjunto de documentos de entrenamiento proporcionado por el usuario.

Muchos de los algoritmos utilizados para la creación de perfiles aprenden un conjunto de características que podrían ayudar a distinguir documentos relevantes de aquellos que no lo son. En base a las ocurrencias de estas características en un nuevo documento, dicho documento será considerado potencialmente útil y mostrado al usuario, o considerado irrelevante y descartado. La mayoría de los sistemas actuales también asignan pesos a las características con el fin de indicar la importancia que tienen en la estimación de la relevancia [35].

Capítulo 3

Modelado Lingüístico Difuso de la Información

En este capítulo vamos a estudiar las distintas técnicas de modelado lingüístico difuso para el manejo de información lingüística, que nos van a proporcionar una mayor flexibilidad en el tratamiento de la información, especialmente en los casos en que se produce una interacción con los usuarios.

3.1. Introducción

La Lógica Difusa se plantea como alternativa a la lógica tradicional, con el objetivo de introducir grados de incertidumbre en las sentencias que califica [100]. Hay numerosas situaciones en las que la lógica tradicional funciona perfectamente. Por ejemplo, supongamos que partimos de las calificaciones obtenidas en una clase y queremos agrupar a los aprobados (aquellos que hayan obtenido una calificación igual o superior a 5). El proceso de razonamiento que se seguiría mediante la lógica tradicional sería ir comparando cada calificación con 5 hasta obtener cuáles están aprobados y cuáles no:

Es cierto que $7 \geq 5$*? SI: Aprobado*

Es cierto que $4 \geq 5$*? NO: No aprobado*

Es cierto que $5 \geq 5$*? SI: Aprobado*

Sin embargo, el inconveniente de esta lógica es que en la vida real no nos encontramos frecuentemente con criterios de clasificación tan tajantes como en el ejemplo. En efecto, hay numerosas situaciones en las que la información no puede ser evaluada cuantitativamente de forma precisa, pero puede que sí sea posible hacerlo cualitativamente, y en estos casos hemos de hacer uso de un *enfoque lingüístico*. Por ejemplo, cuando intentamos cualificar algún fenómeno relacionado con percepciones humanas, a menudo usamos palabras o descripciones en lenguaje natural, en lugar de valores numéricos. Supongamos que dado un conjunto de personas, las intentamos agrupar según su altura. Las personas no son sólo *altas* o *bajas* sino que la mayoría pertenecen a grupos de altura intermedia. La gente suele ser *más bien alta* o *de altura media*. Casi nunca las calificamos con rotundidad, porque el lenguaje que usamos nos permite introducir modificadores que añaden imprecisión: *un poco, mucho, algo...*

Como la lógica tradicional es bivaluada (sólo admite dos valores: o el elemento pertenece al conjunto o no pertenece), se ve maniatada para agrupar según su altura al anterior conjunto de personas, puesto que su solución sería definir un umbral de pertenencia (por ejemplo, un valor que todo el mundo considera que, de ser alcanzado o superado, la persona en cuestión puede llamarse *alta*). Si dicho umbral es 1.80, todas las personas que midan 1.80 o más serán *altas*, mientras que el resto serán bajas. Según esta manera de pensar, alguien que mida 1.79 será tratado igual que otro que mida 1.60, ya que ambos han merecido el calificativo de personas *bajas*.

Si dispusiéramos de una herramienta para caracterizar las alturas de forma que las transiciones entre las que son altas y las que no lo son fueran suaves, estaríamos re-

produciendo la realidad mucho más fielmente. En la realidad hay unos puntos de cruce donde las personas dejan de ser *altas* para ser consideradas *medianas*, de forma que el concepto de *alto* decrece linealmente con la altura. Asignando una función lineal para caracterizar el concepto *alto* en lugar de definir un sólo umbral de separación estamos dando mucha más información acerca de los elementos. Esta función, como veremos, se llamará función de pertenencia.

En este sentido, el uso de la Teoría de Conjuntos Difusos ha dado muy buenos resultados para el tratamiento de información de forma cualitativa [97]. El *modelado lingüístico difuso* es una herramienta que permite representar aspectos cualitativos y que está basada en el concepto de *variables lingüísticas*, es decir, variables cuyo valores no son números, sino palabras o sentencias expresadas en lenguaje natural o artificial [97]. Cada valor lingüístico se caracteriza por un valor sintáctico o *etiqueta* y un valor semántico o *significado*. La etiqueta es una palabra o sentencia perteneciente a un conjunto de términos lingüísticos y el significado es un subconjunto difuso en un universo de discurso.

Se ha demostrado que es una herramienta muy útil en numerosos problemas, como por ejemplo en la toma de decisiones [44, 88, 93], evaluación de la calidad informativa de documentos Web [55], modelos de recuperación de información [15, 50, 51], diagnósticos clínicos [30], análisis político [2], etc.

En este capítulo, vamos a revisar los principales enfoques de modelado lingüístico difuso que podemos usar para el manejo de información lingüística. En la Sección 2 vamos a revisar los conceptos básicos para el manejo de información lingüística. En la Sección 3 vamos a tratar el modelo tradicional, el modelado lingüístico difuso clásico. En la

Sección 4 veremos el modelado lingüístico difuso ordinal definido para eliminar la excesiva complejidad del enfoque lingüístico tradicional. En la Sección 5 nos centraremos en el enfoque lingüístico 2-tupla definido como una mejora del anterior. En la Sección 6 estudiaremos el enfoque lingüístico difuso multi-granular que al permitir trabajar con distintos conjuntos de etiquetas nos será muy útil en aquellos casos en los que no sea eficiente valorar la información usando un mismo sistema de valores. Para finalizar, en la Sección 7 veremos el enfoque lingüístico difuso no balanceado para aplicar en aquellas situaciones en las la información necesite ser valorada sobre un conjunto de etiquetas no uniforme, es decir, asimétrico.

3.2. Conceptos Básicos de Información Lingüística

Vamos a comenzar presentando una revisión de los conceptos básicos de la Teoría de Conjuntos Difusos que van a ser utilizados en el resto de modelados lingüísticos.

El interés de la Teoría de Conjuntos Difusos se centra esencialmente en modelar aquellos problemas donde los enfoques clásicos de la Teoría de Conjuntos y la Teoría de la Probabilidad resultan insuficientes o no operativos. Por ello, generaliza la noción clásica de conjunto e introduce el concepto de *ambigüedad*, de manera que los conjuntos difusos nos proporcionan una nueva forma de representar la imprecisión e incertidumbre presentes en determinados problemas.

3.2.1. Conjuntos Difusos y Funciones de Pertenencia

La noción de conjunto refleja la tendencia a organizar, generalizar y clasificar el conocimiento sobre los objetos del mundo real. El encapsulamiento de los objetos es

una colección cuyos miembros comparten una serie de características o propiedades que implican la noción de conjunto. Los conjuntos introducen una noción de dicotomía, que en esencia es una clasificación binaria: o se acepta o se rechaza la pertenencia de un objeto a una categoría determinada. Habitualmente la decisión de *aceptar* se denota como 1 y la de *rechazar* como 0. Esta decisión de aceptar o rechazar se expresa mediante una función característica, según las propiedades que posean los objetos del conjunto.

La Lógica Difusa se fundamenta en el concepto de ***conjunto difuso*** [98] que suaviza el requerimiento anterior y admite valores intermedios en la función característica, que se denomina ***función de pertenencia***. Esto permite una interpretación más realista de la información, puesto que la mayoría de las categorías que describen los objetos del mundo real, no tienen unos límites claros y bien definidos.

Un conjunto difuso puede definirse como una colección de objetos con valores de pertenencia entre 0 (exclusión total) y 1 (pertenencia total). Los valores de pertenencia expresan los grados con los que cada objeto es compatible con las propiedades o características distintivas de la colección. Formalmente podemos definir un conjunto difuso como sigue.

Definición 3.1. Un ***conjunto difuso*** $\tilde{A}$ sobre un dominio o universo de discurso U está caracterizado por una función de pertenencia que asocia a cada elemento del conjunto el grado con que pertenece a dicho conjunto, asignándole un valor en el intervalo [0,1]:

$$\mu_{\tilde{A}} : U \rightarrow [0, 1]$$

Así, un conjunto difuso $\tilde{A}$ sobre U puede representarse como un conjunto de pares ordenados de un elemento perteneciente a U y su grado de pertenencia, $\tilde{A} = \{(x, \mu_{\tilde{A}}(x))|x \in$

$U, \mu_{\tilde{A}}(x) \in [0,1]\}$. Por ejemplo, consideremos el concepto *persona alta*, en un contexto donde la estatura oscila entre 1 y 2 m. Como es de suponer, alguien que mida 1,30m. no se puede considerar como *persona alta* por lo que su grado de pertenencia al conjunto de personas altas será de 0. Por el contrario, una persona que mida 1,90m. sí la consideramos alta por lo que su grado de pertenencia al conjunto es de 1.

Las gráficas que representan una función de pertenencia pueden adoptar cualquier forma, cumpliendo propiedades específicas, pero es el contexto de la aplicación lo que determina la representación más adecuada en cada caso. Puesto que las valoraciones lingüísticas dadas por los usuarios son únicamente aproximaciones, algunos autores consideran que las funciones de pertenencia trapezoidales lineales son suficientemente buenas para capturar la imprecisión de tales valoraciones lingüísticas. La representación paramétrica es obtenida a partir de una 4-tupla (a, b, α, β), donde a y b indican el intervalo en que el valor de pertenencia es 1, con α y β indicando los límites izquierdo y derecho del dominio de definición de la función de pertenencia trapezoidal. Un caso particular de este tipo de representación son las valoraciones lingüísticas cuyas funciones de pertenencia son triangulares, es decir, $a = b$, por lo que se representan por medio de una 3-tupla (a, α, β). La Figura 3.1 muestra la descripción y la representación gráfica de un ejemplo de función de pertenencia trapezoidal.

3.2.2. Definiciones Básicas

Definición 3.2. Se define el ***soporte*** de un conjunto difuso $\tilde{A}$ en el universo U, como el conjunto formado por todos los elementos de U cuyo grado de pertenencia a $\tilde{A}$ sea mayor que 0:

$$\mu_A(x,a,b,\alpha,\beta)=\begin{cases}0, & x<\alpha\\ \dfrac{x-\alpha}{a-\alpha}, & \alpha\leqslant x\leqslant a\\ 1, & a\leqslant x\leqslant b\\ \dfrac{\beta-x}{\beta-b}, & b\leqslant x\leqslant\beta\\ 0, & x>\beta\end{cases}$$

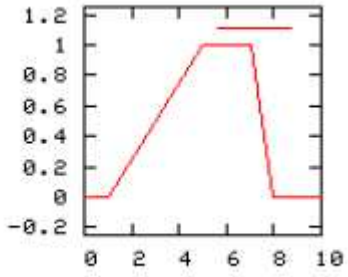

Figura 3.1: Ejemplo de función de pertenencia.

$$supp(\tilde{A}) = \{x \in U \ / \ \mu_{\tilde{A}}(x) > 0\}$$

Definición 3.3. La ***altura*** de un conjunto difuso $\tilde{A}$ se define como el mayor grado de pertenencia de todos los elementos de dicho conjunto:

$$h(\tilde{A}) = max\{\mu_{\tilde{A}}(x) \ / \ x \in U\}$$

Definición 3.4. El α***-corte*** de un conjunto difuso $\tilde{A}$ es el conjunto formado por todos los elementos del universo U cuyos grados de pertenencia en $\tilde{A}$ son mayores o iguales que el valor de corte $\alpha \in [0,1]$:

$$\alpha_{\tilde{A}} = \{x \in U \ / \ \mu_{\tilde{A}}(x) \geq \alpha\}$$

Definición 3.5. Se denomina ***conjunto de niveles*** de un conjunto difuso $\tilde{A}$, al conjunto de grados de pertenencia de sus elementos:

$$L(\tilde{A}) = \{a \ / \ \mu_{\tilde{A}}(x) = a, x \in U\}$$

3.2.3. Operaciones con Conjuntos Difusos

Al igual que en la lógica tradicional, las operaciones lógicas que se pueden establecer entre conjuntos difusos son la intersección, la unión y el complemento. Mientras que el resultado de operar dos conjuntos clásicos es un nuevo conjunto clásico, las mismas operaciones con conjuntos difusos nos darán como resultado otros conjuntos también difusos.

Hay muchas formas de definir estas operaciones. Cualquier operación que cumpla las propiedades de una t-norma puede ser usada para hacer la intersección, de igual manera que cualquier operación que cumpla las propiedades de una t-conorma puede ser empleada para la unión. El cuadro de la Figura 3.2 muestra las propiedades que deben cumplir las dos familias de funciones y algunos ejemplos.

Las operaciones se definen de la siguiente manera:

- Intersección: $\tilde{A} \cap \tilde{B} = \{(x, \mu_{\tilde{A} \cap \tilde{B}}) \ / \ \mu_{\tilde{A} \cap \tilde{B}}(x) = T[\mu_{\tilde{A}}(x), \mu_{\tilde{B}}(x)]\}$
- Unión: $\tilde{A} \cup \tilde{B} = \{(x, \mu_{\tilde{A} \cup \tilde{B}}) \ / \ \mu_{\tilde{A} \cup \tilde{B}}(x) = S[\mu_{\tilde{A}}(x), \mu_{\tilde{B}}(x)]\}$
- Complemento: $\mu_{\sim \tilde{A}}(x) = 1 - \mu_{\tilde{A}}(x)$

En la Figura 3.3 podemos ver una representación gráfica de dichas operaciones.

	Propiedades	Ejemplos
T-Normas **T: [0,1] x [0,1] → [0,1]** $\mu_{A\cap B}(x)=T[\mu_A(x), \mu_B(x)]$	**Conmutativa:** T(a,b)=T(b,a) **Asociativa:** T(a,T(b,c))=T(T(a,b),c) **Monotonía:** T(a,b) > = T(c,d) si a>=c, y b>=d **Condiciones frontera:** T(a,1)=a	**Intersección estándar** T(a,b)=min(a,b) **Producto algebraico** T(a,b)=a · b **Intersección drástica** $T(a,b) = \begin{cases} a, \text{ si } b=1 \\ b, \text{ si } a=1 \\ 0, \text{ en otro caso} \end{cases}$
T-Normas **S: [0,1] x [0,1] → [0,1]** $\mu_{A\cup B}(x)=S[\mu_A(x), \mu_B(x)]$	**Conmutativa:** S(a,b)=S(b,a) **Asociativa:** S(a,S(b,c))=S(S(a,b),c) **Monotonía:** S(a,b) > = S(c,d) si a>=c, y b>=d **Condiciones frontera:** S(a,0)=a	**Unión estándar** S(a,b)=max(a,b) **Suma algebraica** S(a,b)= a + b - a · b **Unión drástica** $S(a,b) = \begin{cases} a, \text{ si } b=0 \\ b, \text{ si } a=0 \\ 1, \text{ en otro caso} \end{cases}$

Figura 3.2: t-normas y t-conormas.

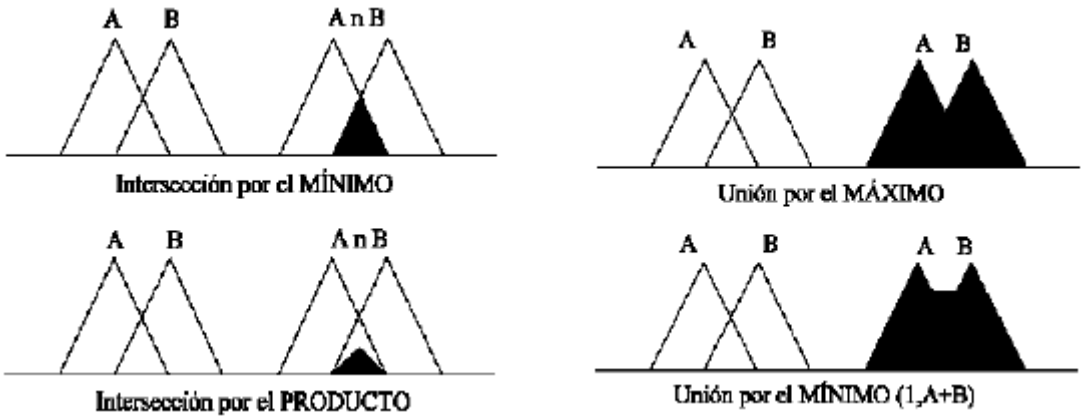

Figura 3.3: Intersección y Unión en conjuntos difusos.

3.2.4. Modelado Lingüístico Difuso

La información que manejamos en el mundo real puede tener diferentes rangos de valoración y los valores pueden tener distinta naturaleza. En ocasiones, puede que no sea fácil valorarla de forma precisa mediante un valor cuantitativo, sin embargo puede que sí sea factible hacerlo de forma cualitativa. En este caso, adoptar un enfoque lingüístico suele ofrecer mejores resultados que si aplicamos uno numérico. Por ejemplo, cuando evaluamos determinados aspectos relacionados con la percepción subjetiva (*diseño, gusto, diversión, etc.*), solemos utilizar palabras en lenguaje natural en lugar de valores numéricos (*bonito, feo, dulce, salado, mucha, poca, etc.*). Esto hecho se puede deber a diversas causas:

- Hay situaciones en las que la información, por su propia naturaleza, no puede ser cuantificada y por tanto únicamente puede ser valorada mediante el uso de términos lingüísticos, como sucede cuando realizamos una valoración sobre un libro que hayamos leído, que solemos usar términos como *bueno*, *regular* o *malo*.

- En otros casos, trabajar con información precisa de forma cuantitativa no es posible, o bien porque no están disponibles los elementos necesarios para llevar a cabo una medición exacta de esa información, o bien porque el coste computacional es demasiado alto y nos basta con la aplicación de un valor aproximado. Por ejemplo, cuando evaluamos la velocidad de una motocicleta, en lugar de usar valores numéricos, solemos usar términos tales como *rápida*, *muy rápida* o *lenta*.

Variables lingüísticas

El modelado lingüístico difuso es, pues, un enfoque aproximado basado en la Teoría de Conjuntos Difusos. Este modelo representa los aspectos cualitativos como valores lingüísticos mediante lo que se conoce como *variables lingüísticas* [97]. Una variable lingüística se caracteriza por un *valor sintáctico* o *etiqueta* que es una palabra o frase perteneciente a un conjunto de términos lingüísticos, y por un *valor semántico* o *significado* de dicha etiqueta que viene dado por un subconjunto difuso en un universo de discurso. Formalmente se define de la siguiente manera.

Definición 3.6. [97] Una ***variable lingüística*** está caracterizada por una 5-tupla $(H, T(H), U, G, M)$, donde:

- H es el nombre de la variable;
- $T(H)$ (o sólo T) simboliza el conjunto de términos lingüísticos de H, es decir, el conjunto de nombres de valores lingüísticos de H, donde cada valor es una variable difusa denotada genéricamente como X que toma valores en el universo de discurso;
- U el universo de discurso que está asociado con una variable base denominada u;
- G es una regla sintáctica (que normalmente toma forma de gramática) para generar los nombre de los valores de H;
- M es una regla semántica para asociar significado a cada elemento de H, que será un subconjunto difuso de U.

Por ejemplo, consideremos la variable lingüística $H = velocidad$, con $U = [0, 125]$ y la variable base $u \in U$. El conjunto de términos asociados con la velocidad podría ser

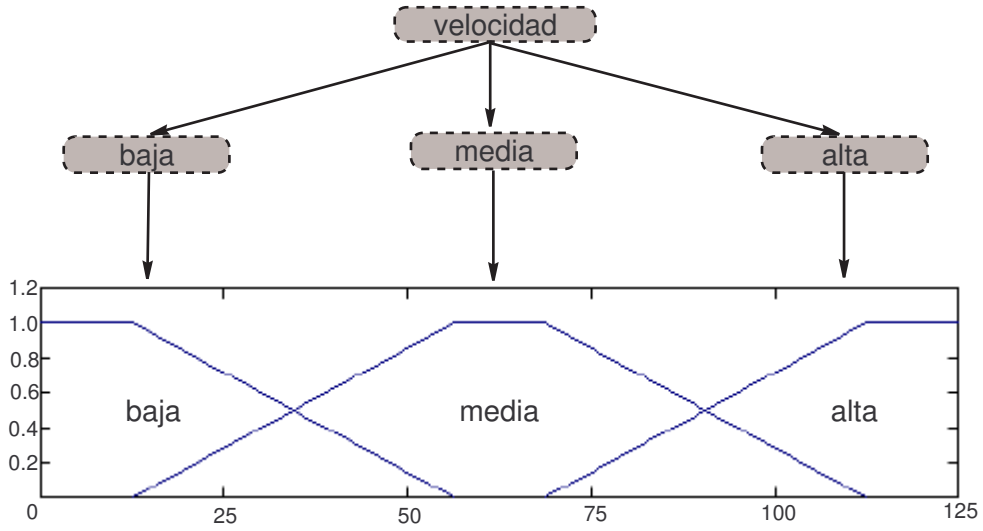

Figura 3.4: Ejemplo de una variable lingüística.

$H(L) = \{baja, media, alta\}$ donde cada término en $H(velocidad)$ es el nombre de un valor lingüístico de *velocidad*. El significado $M(X)$ de una etiqueta $H \in H(velocidad)$ se define como la restricción $H(u)$ sobre la variable base u impuesta según el nombre de H. Por lo tanto $M(X)$ es un conjunto difuso de U cuya función de pertenencia $H(u)$ representa la semántica del nombre H. En la Figura 3.4 podemos ver una representación gráfica del ejemplo.

3.2.5. Pasos para la Aplicación del Enfoque Lingüístico Difuso

En cualquier ámbito en el que deseemos aplicar un enfoque lingüístico para la resolución de algún problema, debemos tomar dos decisiones:

- Modelo de representación: elección del conjunto de términos lingüísticos junto con su semántica y así proporcionar a una fuente de información un número reducido de términos con los que poder expresarla.

- Modelo computacional: definir el modelo computacional seleccionando los correspondientes operadores de comparación y de agregación.

Un aspecto importante que es necesario analizar con el fin de establecer la descripción de una variable lingüística es la ***granularidad de la incertidumbre*** [7], es decir, la cardinalidad del conjunto de términos lingüísticos usado para expresar y representar la información. La cardinalidad debe ser suficientemente baja como para no imponer una precisión excesiva en la información que se quiera expresar y suficientemente alta como para conseguir una discriminación de las valoraciones en un número limitado de grados. Habitualmente la cardinalidad usada en los modelos lingüísticos suele ser un valor impar, como 7 o 9, no superando las 11 o 13 etiquetas. El término medio representa una valoración de *aproximadamente 0.5*, y el resto de términos se sitúan simétricamente alrededor de este punto medio [7]. Estos valores clásicos de cardinalidad están basados en la línea de observación de Miller sobre la capacidad humana [68], en la que se indica que se pueden manejar razonablemente y recordar alrededor de 7 o 9 términos.

Una vez establecida la cardinalidad del conjunto de términos lingüísticos, hay que definir dicho conjunto, es decir, cuáles van a ser las etiquetas lingüísticas y su semántica asociada.

3.3. Modelado Lingüístico Difuso Clásico

El modelado lingüístico difuso clásico adopta un *enfoque basado en una gramática libre de contexto* [7, 13, 97]. Consiste en utilizar una gramática libre de contexto G, donde el conjunto de términos pertenece al lenguaje generado por G. Una gramática genera-

dora G, es una 4-tupla (V_N, V_T, I, P) siendo V_N el conjunto de símbolos no terminales, V_T el conjunto de símbolos terminales, I el símbolo inicial y P el conjunto de reglas de producción. La elección de estos cuatro elementos determinará la cardinalidad y forma del conjunto de términos lingüísticos. Entre los símbolos terminales y no terminales de G podemos encontrar términos primarios (por ejemplo *alto, medio, bajo*), modificadores (por ejemplo *no, mucho, muy, más o menos*), relaciones (por ejemplo *mayor que, menor que*) y conectivos (por ejemplo *y, o, pero*). Siendo I cualquier término primario y usando P, construimos el conjunto de términos lingüísticos $H = \{muy\ alto, alto, medio, ...\}$. La semántica del conjunto de términos lingüísticos se define utilizando números difusos en el intervalo [0,1], dónde cada número difuso es descrito por una función de pertenencia basada en ciertos parámetros o reglas semánticas.

Con respecto a la definición de operadores de agregación de información lingüística, el modelo clásico lo que hace es extender las operaciones de la lógica tradicional para aplicarlas sobre las funciones de pertenencia. El inconveniente es que como resultado obtendremos otro conjunto difuso que no se corresponde con ninguna etiqueta del conjunto de términos originalmente considerado. Si finalmente deseamos obtener una etiqueta de dicho conjunto, es necesario realizar un proceso de aproximación lingüística consistente en encontrar una etiqueta cuyo significado sea el mismo o lo más parecido posible (de acuerdo a alguna métrica) al significado del conjunto difuso no etiquetado obtenido como resultado de alguna operación.

3.4. Modelado Lingüístico Difuso Ordinal

El modelado lingüístico difuso ordinal [31, 42, 44] es un tipo muy útil de enfoque lingüístico difuso, propuesto como una herramienta alternativa al modelado lingüístico

difuso clásico que simplifica la computación con palabras eliminando la complejidad de tener que definir una gramática.

Además, el modelado lingüístico difuso clásico al trabajar con números difusos presenta el inconveniente de que no suelen coincidir con etiquetas del conjunto de términos lingüísticos, por lo que si se desea obtener una etiqueta se hace necesaria una aproximación lingüística. El modelado lingüístico difuso ordinal trabaja directamente con las etiquetas previamente definidas por lo que evita tener que recurrir a aproximaciones lingüísticas complejas.

3.4.1. Modelo de Representación en el Enfoque Lingüístico Ordinal

Un enfoque lingüístico difuso ordinal se define considerando un conjunto de etiquetas finito y totalmente ordenado $\mathcal{S} = \{s_i\}, i \in \{0, \ldots, T\}$ con $s_i \geq s_j$ si $i \geq j$, y con una cardinalidad impar (la cardinalidad de $\mathcal{S}$ es $T + 1$). La semántica del conjunto de etiquetas es establecida según la estructura ordenada del conjunto de etiquetas [16], considerando que cada etiqueta del par (s_i, s_{T-i}) es igualmente informativa. Por ejemplo, podríamos usar el siguiente conjunto de 9 etiquetas para representar la información lingüística:

$$\mathcal{S} = \{N, VL, L, M, H, VH, P\}$$

$$s_0 = Nulo = N \qquad s_1 = Muy\ bajo = VL$$
$$s_2 = Bajo = L \qquad s_3 = Medio = M$$
$$s_4 = Alto = H \qquad s_5 = Muy\ alto = VH$$
$$s_6 = Perfecto = P.$$

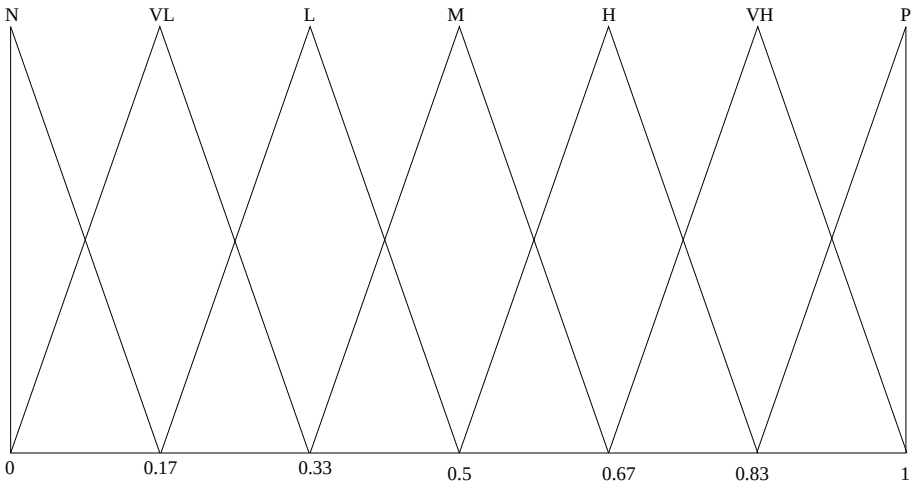

Figura 3.5: Un conjunto de 7 términos lingüísticos y su semántica.

donde $s_a < s_b$ si y sólo si $a < b$.

A continuación, tenemos que dar significado al conjunto de etiquetas lingüísticas asociando con cada término lingüístico un conjunto difuso definido en el intervalo $[0, 1]$. Para ello, podemos hacer uso de una representación trapezoidal de la función de pertenencia, o de su caso más particular, una representación triangular por medio de una 3-tupla (a, α, β), donde recordemos que a indica el punto donde el valor de pertenencia vale 1 y α y β los límites izquierdo y derecho respectivamente. Como ejemplo, podemos considerar el anterior conjunto de etiquetas con las siguientes funciones de pertenencia (ver Figura 3.5):

$$s_0 = Nulo(N) = (0, 0, 0.17) \qquad s_1 = Muy\ bajo(VL) = (0.17, 0, 0.33)$$
$$s_2 = Bajo(L) = (0.33, 0.17, 0.5) \qquad s_3 = Medio(M) = (0.5, 0.33, 0.67)$$
$$s_4 = Alto(H) = (0.67, 0.5, 0.83) \qquad s_5 = Muy\ alto(VH) = (0.83, 0.67, 1)$$
$$s_6 = Perfecto(P) = (1, 0.83, 1).$$

3.4.2. Modelo Computacional en el Enfoque Lingüístico Ordinal

En cualquier enfoque lingüístico necesitamos operadores para el manejo de la información lingüística. Una ventaja del enfoque lingüístico difuso ordinal es la simplicidad y agilidad de su modelo computacional. Está basado en el cálculo simbólico [42, 44] y actúa operando directamente sobre las etiquetas, teniendo en cuenta el orden de las valoraciones lingüísticas en la estructura ordenada de las etiquetas. Habitualmente, el modelo lingüístico difuso ordinal para la computación con palabras se define estableciendo:

1. un operador de negación,
2. operadores de comparación basados en la estructura ordenada de los términos lingüísticos, y
3. operadores apropiados para la agregación de información lingüística difusa ordinal.

En la mayoría de los enfoques lingüísticos difusos ordinales, a partir de la semántica asociada a los términos lingüísticos el operador de negación se define como:

$$NEG(s_i) = s_j \mid j = T - i;$$

También podemos definir dos operadores de comparación de términos lingüísticos:

1. *Operador de maximización*: $MAX(s_i, s_j) = s_i$ si $s_i \geq s_j$.

2. *Operador de minimización*: $MIN(s_i, s_j) = s_i$ si $s_i \leq s_j$.

A partir de estos operadores es posible definir operadores automáticos y simbólicos de agregación de información lingüística, como por ejemplo el operador de agregación de información lingüística no ponderada LOWA (*Linguistic Ordered Weighted Averaging*) [44] y el operador de información lingüística ponderada LWA (*Linguistic Weighted Averaging*) [42], que están basados en el operador OWA (*Ordered Weighted Averaging*) definido en [93]. El OWA es un operador de agregación de información numérica que tiene en cuenta el orden de las valoraciones que van a ser agregadas.

Definición 3.7. *Operador OWA.* Sea $A = \{a_1, ..., a_n\}$ con $a_i \in [0, 1]$ el conjunto de valoraciones que se quieren agregar y $W = (w_1, ..., w_n)$ su vector de pesos asociado, tal que (i) $w_i \in [0, 1]$ y (ii) $\sum_{i=1}^{n} w_i = 1$. El operador OWA, f, se define como:

$$f(a_1, ..., a_n) = \sum_{j=1}^{n} w_j \cdot b_j$$

donde b_j es el j-ésimo mayor valor del conjunto A. Por tanto, a partir de los elementos de A podemos obtener un conjunto B ordenando dichos elementos en orden decreciente, es decir,

$$B = \{b_1, ..., b_n\} \mid b_i \geq b_j \; si \; i < j$$

y definir el operador OWA de la siguiente forma:

$$f(a_1, ..., a_n) = W \cdot B$$

Ejemplo 3.1. Aplicación del operador OWA.

Supongamos que tenemos el siguiente conjunto de valoraciones $A = \{0.6, 1.0, 0.3, 0.5\}$ con el siguiente vector de pesos $W = (0.2, 0.3, 0.1, 0.4)$.

En este caso, el vector ordenado B es

$$B = \begin{bmatrix} 1.0 \\ 0.6 \\ 0.5 \\ 0.3 \end{bmatrix},$$

por lo que:

$$f(0.6, 1.0, 0.3, 0.5) = W \cdot B = [0.2, 0.3, 0.1, 0.4] \begin{bmatrix} 1.0 \\ 0.6 \\ 0.5 \\ 0.3 \end{bmatrix}$$

$$= (0.2 \cdot 1.0) + (0.3 \cdot 0.6) + (0.1 \cdot 0.5) + (0.4 \cdot 0.3) = 0.55$$

Definición 3.8. *Operador LOWA* [44]. Sea $A = \{a_1, \ldots, a_m\}$ un conjunto de etiquetas a agregar, $a_i \in \mathcal{S}$, entonces el operador LOWA, ϕ, se define como:

$$\phi(a_1, \ldots, a_m) = W \cdot B = \mathcal{C}^m\{w_k, b_k, k = 1, \ldots, m\} =$$

$$= w_1 \odot b_1 \oplus (1 - w_1) \odot \mathcal{C}^{m-1}\{\beta_h, b_h, h = 2, \ldots, m\}$$

donde $W = [w_1, \ldots, w_m]$, es un vector de ponderación, tal que,

1. $w_i \in [0, 1]$,

2. $\sum_{i=1}^{n} w_i = 1$,

y $\beta_h = w_h | \Sigma_2^m w_k, h = 2, \ldots, m$, siendo $B = (b_1, \ldots, b_m)$ un vector asociado a A, tal que,

$$B = \sigma(A) = (a_{\sigma(1)}, \ldots, a_{\sigma(n)})$$

donde, $a_{\sigma(j)} \leq a_{\sigma(i)} \ \forall \ i \leq j$, siendo σ una permutación definida sobre el conjunto de etiquetas A. $\mathcal{C}^m$ es el operador de combinación convexa de m etiquetas [32], de modo que si $m = 2$, entonces se define como

$$\mathcal{C}^2\{w_i, b_i, i = 1, 2\} = w_1 \odot s_j \oplus (1 - w_1) \odot s_i = s_k, \quad s_j, \ s_i \ \in \ \mathcal{S}, \ (j \geq i),$$

$$k \ = \ \text{MIN}\{T, i \ + \ round(w_1 \cdot (j - i))\},$$

"*round*" simboliza el operador de redondeo usual, y $b_1 = s_j, \ b_2 = s_i$. Por otro lado, si $w_j = 1$ y $w_i = 0$ con $i \neq j \ \forall i$, entonces el operador de combinación se define como:

$$\mathcal{C}^m\{w_i, b_i, i = 1, \ldots, m\} = b_j.$$

Ejemplo 3.2. Aplicación del operador LOWA.

Supongamos $m = 2$, $W = [0.4, 0.6]$ y que usamos el siguiente conjunto de siete etiquetas:

$$\mathcal{S} = \{s_0 = MI, s_1 = MME, s_2 = ME, s_3 = EQ, s_4 = MA, s_5 = MMA, s_6 = MX\},$$

donde

$MI = Mínimo$ $MME = Mucho\_Menor$ $ME = Menor$

$EQ = Equivalente$ $MA = Mayor$ $MMA = Mucho\_Mayor$

$MX = Máximo$

con los siguiente valores de representación (ver Figura 3.6):

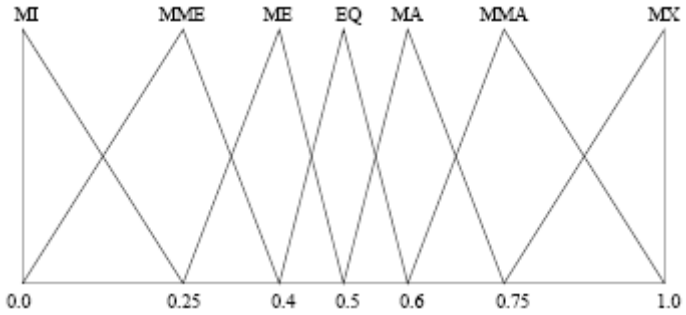

Figura 3.6: Semántica asociada al conjunto de términos lingüísticos.

		1 – w1 = 0.6			
		MME	**MMA**	**MX**	**MME**
w1 = 0.4	**MX**	EQ	MMA	MX	EQ
	MI	MME	EQ	MA	MME
	ME	MME	EQ	MA	MME
	EQ	ME	MA	MMA	ME

Figura 3.7: Tabla del LOWA con $m = 2$.

$MI = (0, 0, 0, 0.25)$ $MME = (0.25, 0.25, 0, 0.4)$ $ME = (0.4, 0.4, 0.25, 0.5)$

$EQ = (0.5, 0.5, 0.4, 0.6)$ $MA = (0.6, 0.6, 0.5, 0.75)$ $MMA = (0.75, 0.75, 0.6, 1)$

$MX = (1, 1, 0.75, 1)$

Los resultados se muestran en la Tabla 3.7, donde por ejemplo:

$$k_{11} = \text{MIN}\{6, 1 + round(0.4 * (6 - 1))\} = 3 \Rightarrow l_{k_{11}} = \text{EQ}$$
$$k_{21} = \text{MIN}\{6, 0 + round(0.6 * (1 - 0))\} = 1 \Rightarrow l_{k_{21}} = \text{MME}$$

Para concluir, indicar que existen otras opciones de modelado lingüístico difuso or-

dinal, como generar la semántica de las etiquetas lingüísticas utilizando funciones de negación que inducen una semántica para cada etiqueta [87], estando éstas definidas como intervalos en [0,1].

3.5. Modelado Lingüístico Difuso 2-tupla

El *modelado lingüístico difuso 2-tupla* [47, 48] es un tipo de modelado lingüístico difuso que nos permite reducir la pérdida de información que habitualmente se produce en el modelado lingüístico difuso ordinal. Esta pérdida de información, que provoca una falta de precisión en los resultados, se debe al propio modelo de representación puesto que opera con valores discretos sobre un universo de discurso continuo. La principal ventaja del modelo computacional lingüístico basado en 2-tupla, es que permite realizar procesos de cómputo con palabras de forma más precisa y por tanto, sin pérdida de información puesto que utiliza un modelo continuo de representación de la información. Para definirlo, tenemos que establecer el modelo de representación y el modelo computacional para representar y agregar la información lingüística respectivamente.

3.5.1. Modelo de Representación en el Enfoque Lingüístico 2-tupla

Consideremos que $\mathcal{S} = \{s_0, ..., s_T\}$ es un conjunto de términos lingüísticos con cardinalidad impar, donde el término intermedio representa una valoración de aproximadamente 0.5 y con el resto de términos del conjunto distribuidos simétricamente alrededor de ese punto intermedio. Asumimos que la semántica asociada con cada una de las etiquetas viene dada por medio de funciones de pertenencia triangulares, representadas por 3-tuplas (a, α, β) y consideramos todos los términos distribuidos sobre una escala

sobre la que hay establecida una relación de orden total, es decir, $s_i \leq s_j \iff i \leq j$. En este contexto lingüístico difuso, si mediante un método simbólico de agregación de información lingüística [42, 44] obtenemos un valor $\beta \in [0, T]$, y $\beta \notin \{0, ..., T\}$, podemos usar una función de aproximación para expresar el resultado obtenido como un valor de $\mathcal{S}$.

Definición 3.9. [47] Sea β el resultado de una agregación de los índices de un conjunto de etiquetas valoradas sobre un conjunto de términos lingüísticos $\mathcal{S}$, es decir, el resultado de una operación de agregación simbólica, $\beta \in [0, T]$. Dados $i = round(\beta)$ y $\alpha = \beta - i$ dos valores, tales que, $i \in [0, T]$ y $\alpha \in [-0.5, 0.5)$ entonces α es lo que denominamos ***Traslación Simbólica***, que expresa la diferencia de información entre la información expresada por β y la etiqueta lingüística s_i más cercana a $\mathcal{S}$.

El enfoque lingüístico difuso basado en 2-tupla se desarrolla a partir del concepto de traslación simbólica, representando la información lingüística por medio de una 2-tupla (s_i, α_i), $s_i \in \mathcal{S}$ y $\alpha_i \in [-0.5, 0.5)$:

- s_i representa la etiqueta lingüística, y
- α_i es un valor numérico que expresa la traslación de β al índice de la etiqueta más cercana, i, en el conjunto de términos lingüísticos ($s_i \in \mathcal{S}$).

Este modelo define un conjunto de funciones de transformación entre valores numéricos y 2-tupla.

Definición 3.10. Sea $s_i \in \mathcal{S}$ un término lingüístico, su representación mediante una 2-tupla se obtiene mediante la función θ:

$$\theta : [0, T] \longrightarrow \mathcal{S} \times [-0.5, 0.5)$$

$$\theta(s_i) = (s_i, 0) | s_i \in \mathcal{S}$$

Definición 3.11. [47] Siendo $\mathcal{S} = \{s_0, ..., s_T\}$ un conjunto de términos lingüísticos y $\beta \in [0, T]$ un valor que representa el resultado de una operación de agregación simbólica, la 2-tupla que expresa la información equivalente a β se obtiene mediante la siguiente función:

$$\Delta : [0, T] \longrightarrow \mathcal{S} \times [-0.5, 0.5)$$

$$\Delta(\beta) = (s_i, \alpha), \ con \begin{cases} s_i & i = round(\beta) \\ \alpha = \beta - i & \alpha \in [-0.5, 0.5) \end{cases}$$

donde $round(\cdot)$ es el típico operador de redondeo, s_i es la etiqueta cuyo índice es el más cercano a β y α es el valor de la traslación simbólica.

Ejemplo 3.3. Representación 2-tupla.

Supongamos que trabajamos con el siguiente conjunto de términos lingüísticos $\mathcal{S} = \{s_0, s_1, s_2, s_3, s_4, s_5, s_6\}$ y que como resultado de una operación de agregación simbólica se obtiene el valor $\beta = 2.8$. La representación de este valor mediante una 2-tupla lingüística, sería:

$$\Delta(\beta) = (s_3, -0.2)$$

Definición 3.12. [47] Sea $\mathcal{S} = \{s_0, ..., s_T\}$ un conjunto de términos lingüísticos y (s_i, α) una 2-tupla. Se define la función Δ^{-1} , tal que aplicada sobre una 2-tupla (s_i, α) devuelve su valor numérico $\beta \in [0, T]$.

$$\Delta^{-1} : \mathcal{S} \times [-0.5, 0.5) \longrightarrow [0, T]$$

$$\Delta^{-1}(s_i, \alpha) = i + \alpha = \beta$$

3.5.2. Modelo Computacional en el Enfoque Lingüístico 2-tupla

A continuación presentamos el modelo computacional que nos permite operar sobre la representación lingüística 2-tupla, basándonos en los operadores de comparación, negación y agregación de 2-tupla:

1. *Operador de comparación 2-tupla.* La comparación de información lingüística representada por medio de 2-tupla se realiza de acuerdo a un orden lexicográfico normal y corriente. Consideremos dos 2-tupla (s_k, α_1) y (s_l, α_2) que representan cantidades de información:

   - si $k < l$ entonces (s_k, α_1) es menor que (s_l, α_2).
   - si $k = l$ entonces

     a) si $\alpha_1 = \alpha_2$ entonces (s_k, α_1) y (s_l, α_2) representan la misma información,

     b) si $\alpha_1 < \alpha_2$ entonces (s_k, α_1) es menor que (s_l, α_2),

c) si $\alpha_1 > \alpha_2$ entonces (s_k, α_1) es mayor que (s_l, α_2).

2. *Operador de negación 2-tupla.* El operador de negación sobre una 2-tupla se define como:

$$Neg((s_i, \alpha)) = \Delta(T - (\Delta^{-1}(s_i, \alpha))).$$

siendo $T + 1$ la cardinalidad del conjunto de etiquetas $\mathcal{S}$.

3. *Operador de agregación 2-tupla.* La agregación de información consiste en obtener un valor que resuma un conjunto de valores, por lo que el resultado de la agregación de un conjunto de varias 2-tupla debe ser una 2-tupla. En la literatura podemos encontrar numerosos operadores de agregación que nos permiten combinar la información de acuerdo a distintos criterios. Cualquiera de estos operadores puede ser fácilmente extendido para trabajar con 2-tupla, usando funciones Δ y Δ^{-1} que transforman valores numéricos en 2-tupla y viceversa sin pérdida de información. Algunos ejemplos de estos operadores son los siguientes:

Definición 3.13. *Media aritmética.* Siendo $x = \{(r_1, \alpha_1), \ldots, (r_n, \alpha_n)\}$ un conjunto de varias 2-tupla lingüísticas, la 2-tupla que simboliza la media aritmética, $\overline{x}^e$, se calcula de la siguiente forma:

$$\overline{x}^e[(r_1, \alpha_1), \ldots, (r_n, \alpha_n)] = \Delta(\sum_{i=1}^{n} \frac{1}{n} \Delta^{-1}(r_i, \alpha_i)) = \Delta(\frac{1}{n} \sum_{i=1}^{n} \beta_i).$$

Definición 3.14. *Operador de media ponderada.* Siendo $x = \{(r_1, \alpha_1), \ldots, (r_n, \alpha_n)\}$ un conjunto de varias 2-tupla lingüísticas y $W = \{w_1, ..., w_n\}$ un vector numérico

con sus pesos asociados, la 2-tupla que simboliza la media ponderada, $\overline{x}^w$, es:

$$\overline{x}^w[(r_1,\alpha_1),\ldots,(r_n,\alpha_n)] = \Delta(\frac{\sum_{i=1}^{n}\Delta^{-1}(r_i,\alpha_i)\cdot w_i}{\sum_{i=1}^{n} w_i}) = \Delta(\frac{\sum_{i=1}^{n}\beta_i\cdot w_i}{\sum_{i=1}^{n} w_i}).$$

Definición 3.15. *Operador de media ponderada lingüística.* Siendo $x = \{(r_1,\alpha_1), \ldots,(r_n,\alpha_n)\}$ un conjunto de varias 2-tupla y $W = \{(w_1,\alpha_1^w),...,(w_n,\alpha_n^w)\}$ sus pesos asociados representados mediante 2-tupla lingüísticas, la 2-tupla que representa la media ponderada lingüística, $\overline{x}_l^w$, se calcula de la siguiente manera:

$$\overline{x}_l^w[((r_1,\alpha_1),(w_1,\alpha_1^w))...((r_n,\alpha_n),(w_n,\alpha_n^w))] = \Delta(\frac{\sum_{i=1}^{n}\beta_i\cdot\beta_{W_i}}{\sum_{i=1}^{n}\beta_{W_i}}),$$

con $\beta_i = \Delta^{-1}(r_i,\alpha_i)$ y $\beta_{W_i} = \Delta^{-1}(w_i,\alpha_i^w)$.

3.6. Modelado Lingüístico Difuso Multi-granular

Con anterioridad hemos comentado que en cualquier enfoque lingüístico difuso, uno de los parámetros más importantes que hay que determinar es la *granularidad de la incertidumbre*, es decir, la cardinalidad del conjunto de términos lingüísticos $\mathcal{S}$ usado para expresar la información lingüística. En función del grado de incertidumbre que un experto encargado de cualificar un fenómeno tenga sobre el mismo, el conjunto de términos lingüísticos elegido para proporcionar ese conocimiento tendrá más o menos términos. Por lo tanto, cuando distintos expertos tienen diferentes grados de incertidumbre sobre el fenómeno, es conveniente que cada uno trabaje con conjuntos de términos lingüísticos de diferente granularidad de incertidumbre (es decir, trabajar con información lingüística multi-granular) [43, 49, 54]. El uso de diferentes conjuntos de

etiquetas es también necesario cuando un experto tiene que valorar conceptos diferentes, como por ejemplo ocurre en los problemas de recuperación de información, al evaluar la importancia de los términos de la consulta y la relevancia de los documentos recuperados [53], que son conceptos distintos. En ese tipo de situaciones necesitamos herramientas que nos permitan gestionar información lingüística multi-granular, es decir, necesitamos definir un *modelado lingüístico difuso multi-granular*. Para ello vamos a seguir el modelo propuesto en [49] que hace uso del concepto de jerarquías lingüísticas.

Una ***Jerarquía Lingüística*** es un conjunto de niveles, donde cada nivel, a su vez, es un conjunto de términos lingüísticos con una granularidad diferente del resto de niveles de la jerarquía [26]. A cada uno de los niveles de una jerarquía lingüística los vamos a denotar como $l(t, n(t))$, siendo t un número que indica el nivel de la jerarquía y $n(t)$ la granularidad del conjunto de términos lingüísticos del nivel t.

Normalmente, las jerarquías lingüísticas trabajan con términos lingüísticos cuyas funciones de pertenencia son de forma triangular, simétricas y uniformemente distribuidas en el intervalo [0,1]. Además, los conjuntos de términos lingüísticos tienen una granularidad impar, con la etiqueta central indicando un valor de indiferencia.

Los niveles de una jerarquía lingüística están ordenados en función de su granularidad, es decir, que para dos niveles consecutivos t y $t+1$, $n(t+1) > n(t)$. Por lo tanto, cada nivel $t+1$ proporciona un refinamiento lingüístico con respecto al nivel anterior t.

Vamos a definir una jerarquía lingüística, LH, como la unión de todos los niveles t que

la conforman:

$$LH = \bigcup_t l(t, n(t)).$$

Para la construcción de LH debemos tener en mente que el orden jerárquico nos viene dado por el incremento de granularidad de los conjuntos de términos lingüísticos de cada nivel.

Partiendo de que $\mathcal{S}^{n(t)} = \{s_0^{n(t)}, ..., s_{n(t)-1}^{n(t)}\}$ sea el conjunto de términos lingüísticos definido para el nivel t con $n(t)$ términos, la construcción de una jerarquía lingüística debe satisfacer las siguientes reglas básicas [49]:

1. Preservar todos los puntos modales previos de las funciones de pertenencia de cada uno de los términos lingüísticos de cada nivel con respecto a los del nivel siguiente.

2. Hacer que las transacciones entre dos niveles consecutivos sean suaves. El propósito es construir un nuevo conjunto de términos lingüísticos, $\mathcal{S}^{n(t+1)}$, de forma que añadiremos un nuevo término lingüístico entre cada pareja de términos pertenecientes al conjunto de términos del nivel anterior t. Para realizar esta inserción de nuevos términos, reduciremos el soporte de las etiquetas lingüísticas para dejar hueco entre ellas para la nueva etiqueta.

De forma genérica, podemos establecer que el conjunto de términos lingüísticos de nivel $t + 1$, $\mathcal{S}^{n(t+1)}$, puede ser obtenido a partir del nivel anterior t, $\mathcal{S}^{n(t)}$, de la siguiente manera:

$$l(t, n(t)) \rightarrow l(t + 1, 2 \cdot n(t) - 1)$$

En el cuadro de la Tabla 3.8 mostramos la granularidad necesaria en cada conjunto de términos lingüísticos de nivel t, dependiendo del valor $n(t)$ definido en el primer nivel (para valores de 3 y 7 respectivamente).

	Nivel 1	Nivel 2	Nivel 3
l(t,n(t))	l(1,3)	l(2,5)	l(3,9)
l(t,n(t))	l(1,7)	l(2,13)	

Figura 3.8: Granularidad en distintos niveles de una jerarquía.

En la Figura 3.9 se muestra un ejemplo gráfico de jerarquías lingüísticas. Se representa una jerarquía compuesta de 3 niveles, de 3, 5 y 9 etiquetas cada uno de ellos.

En [49] se demostró que las jerarquías lingüísticas son útiles para representar información lingüística multi-granular y por tanto permiten trabajar con información lingüística sin pérdida de información. Para conseguirlo, fue definida una familia de funciones de transformación entre etiquetas de diferentes niveles.

Definición 3.16. . Sea $LH = \bigcup_t l(t, n(t))$ una jerarquía lingüística cuyos conjuntos de términos lingüísticos son denotados como $\mathcal{S}^{n(t)} = \{s_0^{n(t)}, ..., s_{n(t)-1}^{n(t)}\}$. La ***función de transformación*** de una etiqueta lingüística (representada mediante una 2-tupla) de un nivel t a una etiqueta de un nivel consecutivo $t + c$, con $c \in -1, 1$, se define como:

$$\tau_{t+c}^{t} : l(t, n(t)) \longrightarrow l(t + c, n(t + c))$$

$$\tau_{t+c}^{t}(s_i^{n(t)}, \alpha^{n(t)}) = \Delta(\frac{\Delta^{-1}(s_i^{n(t)}, \alpha^{n(t)}) \cdot (n(t + c) - 1)}{n(t) - 1})$$

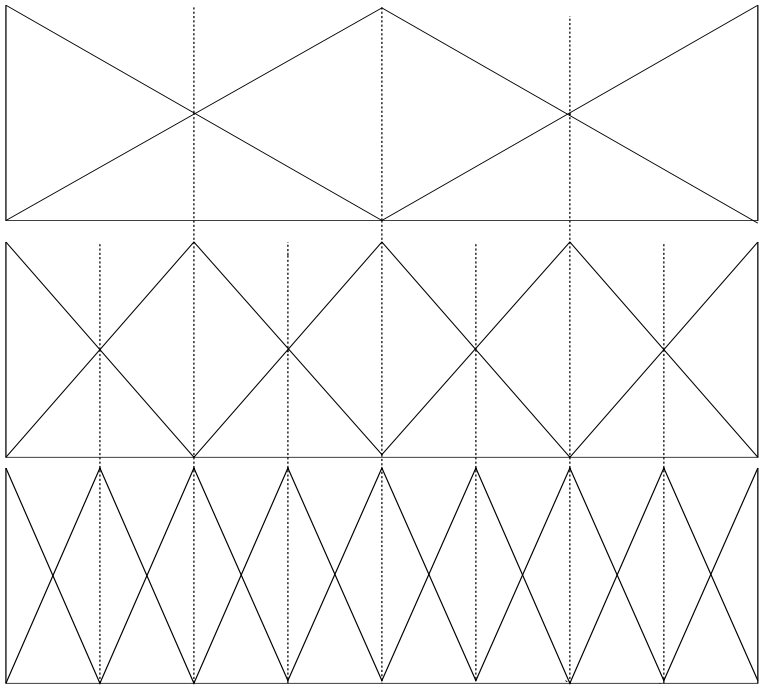

Figura 3.9: Jerarquía lingüística de 3, 5 y 9 etiquetas.

Esta función de transformación fue generalizada para transformar términos lingüísticos entre cualquier nivel dentro de la jerarquía lingüística.

Definición 3.17. Sea $LH = \bigcup_t l(t, n(t))$ una jerarquía lingüística cuyos conjuntos de términos lingüísticos son denotados como $\mathcal{S}^{n(t)} = \{s_0^{n(t)}, ..., s_{n(t)-1}^{n(t)}\}$. La ***función de transformación recursiva*** entre una etiqueta lingüística (representada mediante una 2-tupla) perteneciente a un nivel t y una etiqueta perteneciente al nivel $t' = t + a$, con $a \in Z$, se define como:

$$\tau_{t'}^{t} : l(t, n(t)) \longrightarrow l(t', n(t'))$$

Si $|a| > 1$ entonces

$$\tau_{t'}^{t}(s_i^{n(t)}, \alpha^{n(t)}) = \tau_{t'}^{t+\frac{t-t'}{|t-t'|}}(\tau_{t+\frac{t-t'}{|t-t'|}}^{t}(s_i^{n(t)}, \alpha^{n(t)}))$$

Si $|a| = 1$ entonces

$$\tau_{t'}^{t}(s_i^{n(t)}, \alpha^{n(t)}) = \tau_{t+\frac{t-t'}{|t-t'|}}^{t}(s_i^{n(t)}, \alpha^{n(t)})$$

Esta función de transformación recursiva, puede ser definida fácilmente de una forma no recursiva de la siguiente manera:

$$\tau_{t'}^{t} : l(t, n(t)) \longrightarrow l(t', n(t'))$$

$$\tau_{t'}^{t}(s_i^{n(t)}, \alpha^{n(t)}) = \Delta(\frac{\Delta^{-1}(s_i^{n(t)}, \alpha^{n(t)}) \cdot (n(t') - 1)}{n(t) - 1})$$

Proposición 2.1 [49]. Esta familia de funciones de transformación entre etiquetas lingüísticas de distintos niveles de una jerarquía lingüística es biyectiva:

$$\tau_{t}^{t'}(\tau_{t'}^{t}(s_i^{n(t)}, \alpha^{n(t)})) = (s_i^{n(t)}, \alpha^{n(t)})$$

3.7. Modelos de Sistemas de Recuperación de Información basados en Modelado Lingüístico

El modelado lingüístico difuso se ha mostrado como una herramienta muy útil en numerosos problemas, como por ejemplo en la toma de decisiones [44, 88, 93], evaluación de la calidad informativa de documentos Web [55], diagnósticos clínicos [30], análisis político [2], etc.

También, el modelado lingüístico difuso ha sido aplicado al ámbito de la RI, como muestran los trabajos [13, 15, 50, 51, 53]. Estos trabajos tratan de modelar, de manera lingüística, la representación de las necesidades de información de los usuarios, para lo cual desarrollan un lenguaje de consulta, y un mecanismo de evaluación asociado.

Así por ejemplo, en [13] Bordogna et al. abordan este tema, centrándose en describir las consultas Booleanas ponderadas con pesos lingüísticos. Para introducir requerimientos cualitativos en las consultas Booleanas ponderadas, remplaza los pesos numéricos por descriptores lingüísticos de los documentos deseados. En este trabajo los descriptores lingüísticos juegan el papel de especificaciones difusas de los pesos del términos índice ideales. Las valoraciones lingüísticas son usadas para transformar los valores numéricos en valor difusos. La principal limitación de este modelo de SRI es que no es del todo lingüístico, ya que a pesar de que el usuario introduce valoraciones lingüísticas para establecer los pesos de la consulta ponderada, el sistema obtiene los RSV de los documentos valorados numéricamente.

En [15] Bordogna et al. proponen un modelo de SRI donde ahora sí, tanto los pesos de los términos de la consulta y los RSV obtenidos por el sistema son valoraciones lingüísticas.

Ambos trabajos contemplan sólo un elemento de ponderación, los términos de la consulta, y sólo una posibilidad semántica, semántica del documento perfecto, o documento ideal. En [50] Herrera-Viedma propone un modelo de SRI basado en el modelo lingüístico ordinal de representación de información. Este modelo permitía dos niveles de ponderación, a nivel de términos y a nivel de subexpresiones, y dos posibilidades semánticas, una de umbral a nivel de términos y otra de importancia relativa a nivel de

subexpresiones. También Herrera-Viedma en [51] propone un nuevo modelo de SRI ordinal, con un único elemento de ponderación (términos) y cuya principal característica es contar con un subsistema de consulta y evaluación que permitía al usuario utilizar tres posibilidades semántica, incluso de manera simultánea. Adicionalmente, en [50] y [51] se proponen dos posibles interpretaciones de la semántica de umbral simétrico[1] .

Más recientemente, en [53] se ha propuesto un nuevo modelo de SRI lingüístico ordinal multi-granular. Este sistema permite el uso de diferentes conjuntos de etiquetas, incluso con diferente granularidad para expresar las distintas ponderaciones semánticas. En él se pueden usar hasta cuatro conjuntos distintos de etiquetas, cada uno de ellos con distinta semántica. Tal modelo también describe una metodología para agregar esta información multi-granular.

[1] La interpretación semántica de umbral simétrico y el modelo de SRI lingüístico ordinal propuestos en [51] serán descritos en detalle en la Sección 4.2.

Capítulo 4

Un Nuevo Modelo de Sistema de Recuperación de Información Basado en 2-tupla

Los SRI basados en el enfoque lingüístico difuso ordinal presentan algunos problemas de pérdida de precisión e información asociados al hecho de trabajar con dominios de expresión lingüísticos discretos y utilizar métodos de agregación simbólica y deficiencias en el proceso de evaluación de consultas. En este capítulo presentaremos un nuevo modelo de SRI basado en el enfoque lingüístico difuso 2-tupla que nos permite superar los problemas de los SRI lingüísticos ordinales y mejorar su rendimiento.

4.1. Introducción

La principal actividad de un SRI, como se comentó en la Sección 2.1, es la obtención de documentos relevantes que mejor satisfagan las consultas de los usuarios. Como también hemos comentado en la Sección 2.3, estos sistemas suelen tener tres componentes: una *base de datos*, un *subsistema de consulta* y un *subsistema de evaluación*.

El subsistema de consulta soporta la interacción entre el usuario y el sistema de recuperación de información, y por tanto, debería tener en cuenta la imprecisión y vaguedad típica de la comunicación humana. Este aspecto puede ser modelado introduciendo ponderaciones en el lenguaje de consulta. Muchos autores han propuesto modelos de SRI ponderados usando la teoría de conjunto difuso [8, 10, 14, 19, 20, 21, 62, 91, 92, 95]. Generalmente, estos autores asumen pesos numéricos asociados con las consultas (valores en [0,1]). Sin embargo, el uso de lenguajes de consulta basados en pesos numéricos fuerza al usuario a cuantificar conceptos cualitativos (tales como "importancia"), ignorando que muchos usuarios no son capaces de proporcionar sus necesidades de información de manera precisa en forma cuantitativa, pero si en forma cualitativa. De hecho, parece más natural caracterizar el contenido de los documentos asociando de manera explícita un descriptor lingüístico a los términos de la consulta, como "importante" o "muy importante", en vez de un valor numérico. En este sentido, algunos modelos de SRI [13, 15, 50, 51, 53, 61] han sido propuestos usando un enfoque lingüístico difuso [97] para modelar los pesos de la consulta y la relevancia de los documentos.

Un enfoque lingüístico difuso que nos permite reducir la complejidad del diseño de estos SRI [50, 51] es el enfoque lingüístico difuso ordinal [15, 42, 45, 94]. En este enfoque, los pesos de la consulta y la relevancia de los documentos son términos lingüísticos ordinales [15, 50, 51]. Estos modelos son afectados por los dos problemas característicos del modelado lingüístico difuso ordinal [46]:

- *La pérdida de precisión*: El enfoque lingüístico difuso ordinal trabaja con dominios lingüísticos discretos y esto implica algunas limitaciones en la representación de la información lingüística, por ejemplo, para representar los grados de relevancia.

- *La pérdida de información*: Los operadores de agregación de información lingüística ordinal usan operaciones de aproximación en sus definiciones (por ejemplo la operación de redondeo), y por tanto, presentan pérdidas de información en los procesos de agregación lingüística.

Existen en las consultas cuatro elementos de ponderación posibles: términos, subexpresiones, conectores y consulta completa; y cuatro posibles interpretaciones semánticas: *semántica de umbral* (que considera los pesos de la consulta como requerimientos a satisfacer por cada término en la consulta en el momento de emparejar la consulta y la representación de los documentos), *semántica de perfección* (define los pesos de la consulta como descripciones del documento perfecto deseado por el usuario), *semántica cuantitativa* (los pesos de la consulta expresan la cantidad de documentos a recuperar), and *semántica de importancia* (que considera los pesos como medida de la importancia relativa de cada término de la consulta con respecto a los demás).

En [51] se presentó un modelo de SRI lingüístico difuso ordinal que acepta consultas ponderadas basadas sólo en un elemento de ponderación (términos de la consulta) y permite asociar diferentes interpretaciones semánticas a los pesos. Este sistema usa la t-conorma MIN and t-norma MAX como operadores para evaluar los conectores lógicos Booleanos AND y OR en el proceso de recuperación de información. Este modelo presenta las anteriores limitaciones, y por tanto, los problemas derivados del uso de los operadores MIN y MAX, esto es, la pérdida de flexibilidad en el cálculo de los RVSs de los documentos. Además presentaba funciones de evaluación de bajo rendimiento que no se adecúan al significado dado a las consultas ponderadas por los usuarios.

El principal objetivo de este capítulo es presentar un nuevo modelo de SRI lingüísti-

co difuso, diseñado usando el enfoque lingüístico difuso 2-tupla [46] para superar las principales limitaciones del modelo de SRI lingüístico difuso ordinal definido en [51]. Además, introducimos un nuevo operador de soft computing para modelar los conectores Booleanos en una forma flexible, el operador lingüístico LOWA 2-tupla, así como una nueva interpretación de la semántica de umbral simétrico introducida en [50, 51]. Esta nueva interpretación de la semántica de umbral simétrico también se beneficiará de las bondades del modelado lingüístico difuso 2-tupla. Con todo ésto, el rendimiento del modelo de SRI lingüístico ordinal es mejorado con un costo limitado.

Este capítulo se estructura como sigue. En la Sección 4.2 introducimos el modelo de SRI lingüístico difuso basado en el enfoque lingüístico ordinal definido en [51]. El nuevo modelo de SRI lingüístico difuso basado en el enfoque lingüístico 2-tupla se presenta en la Sección 4.3, junto con un ejemplo teórico y otro práctico de su rendimiento. En la Sección 4.4 propondremos una nueva función de evaluación de términos ponderados, junto con algunos ejemplos. Por último, presentamos algunas reflexiones sobre este nuevo modelo de SRI lingüístico difuso.

4.2. Un Sistema de Recuperación de Información Lingüístico Difuso Ordinal

En [51], se presentó un modelo de SRI ponderado lingüístico ordinal que presenta los siguientes elementos para llevar a cabo su actividad:

Base de datos. La base de datos almacena el conjunto finito de documentos $\mathcal{D} = \{d_1, \ldots, d_m\}$ representados por un conjunto finito de términos índice $\mathcal{T} = \{t_1, \ldots, t_l\}$, que describen el contenido subyacente de los documentos. La representación de

un documento es un conujunto difuso de términos caracterizado por una función numérica de indexación $\mathcal{F} : \mathcal{D} \times \mathcal{T} \rightarrow [0, 1]$, que se llama función de ponderación de términos índice [91]:

$$d_j = \mathcal{F}(d_j, t_1)/t_1 + \mathcal{F}(d_j, t_2)/t_2 + \ldots + \mathcal{F}(d_j, t_l)/t_l.$$

$\mathcal{F}$ pondera los términos índice conforme a su importancia en describir el contenido de un documento. De esta manera $\mathcal{F}(d_j, t_i)$ es un peso numérico que representa el grado importancia de t_i en d_j.

Subsistema de Consulta. El subsistema de consulta presenta un lenguaje de consulta Booleano ponderado para expresar las necesidades de información del usuario. Con este lenguaje cada consulta se expresa como una combinación de términos índice ponderados conectados por operadores lógicos AND ($\wedge$), OR ($\vee$), y NOT ($\neg$). Cada término en la consulta puede ser ponderado de acuerdo a tres diferentes posibilidades semánticas, incluso de manera simultánea. Como en [13], usamos la variable lingüística *Importancia* para expresar los pesos lingüísticos asociados a los términos de la consulta. De esta manera, consideramos un conjunto de valores lingüísticos ordinales $\mathcal{S}$ para expresar los pesos lingüísticos. Definimos una consulta Booleana ponderada como cualquier expresión Booleana legítima cuyos componentes atómicos (átomos) son 4-tuplas $\langle t_i, c_i^1, c_i^2, c_i^3 \rangle$ pertenecientes al conjunto, $\mathcal{T} \times \mathcal{S}^3, t_i \in \mathcal{T}$, y c_i^1, c_i^2, c_i^3 son valores ordinales de la variable lingüística *Importancia*, modelando una semántica de umbral, una semántica cuantitativa, y una semántica de importancia relativa, respectivamente. Cosecuentemente, el conjunto $\mathcal{Q}$ de consultas legítimas se define con las siguientes reglas sintácticas:

1. $\forall q = \langle t_i, c_i^1, c_i^2, c_i^3 \rangle \in \mathcal{T} \times \mathcal{S}^3 \rightarrow q \in \mathcal{Q}$.
2. $\forall q, p \in \mathcal{Q} \rightarrow q \wedge p \in \mathcal{Q}$.

3. $\forall q, p \in \mathcal{Q} \rightarrow q \vee p \in \mathcal{Q}$.

4. $\forall q \in \mathcal{Q} \rightarrow q \in \mathcal{Q}$.

5. Todas las consultas legítimas $q \in \mathcal{Q}$ son sólo aquellas obtenidas aplicando las reglas 1-4, inclusive.

Subsistema de Evaluación. El objetivo del subsistema de evaluación consiste en evaluar documentos en términos de su relevancia a consultas Booleanas ponderadas lingüísticas de acuerdo a tres posibilidades semánticas. Un consulta Booleana con más de un término ponderado se evalúa por medio de un proceso constructivo ascendente basado en el criterio de separabilidad [21, 91]. Este proceso incluye los siguientes cinco pasos:

1. *Preprocesamiento de la consulta:* En este paso, la consulta del usuario es preprocesada y transformada en otra en forma normal conjuntiva (CNF) o en forma normal disyuntiva (DNF), con el resultado de que todas sus subexpresiones Booleanas quedan con al menos dos átomos.

2. *Evaluación de los átomos con respecto a la semántica de umbral:* En este paso, los documentos son evaluados con respecto a su relevancia a átomos individuales en la consulta, considerando sólo las restricciones impuestas por la semántica de umbral. En [51] los documentos eran evaluados usando una semántica de umbral simétrico. Según esta semántica, un usuario puede buscar documentos con una mínima presencia aceptable de un término en su representación, o documentos con una máxima presencia aceptable de un término en su representación [51]. Esto es representado por la siguiente

función de evaluación paramétrica:

$$RSV_j^{i,1} = g^1(d_j, t_i, c_i^1) = \begin{cases} s_0 & s_b \geq s_{\frac{T}{2}} \wedge s_a = s_0 \\ s_{i_1} & s_b \geq s_{\frac{T}{2}} \wedge s_0 < s_a < s_b \\ s_{i_2} & s_b \geq s_{\frac{T}{2}} \wedge s_b \leq s_a < s_T \\ s_T & s_b \geq s_{\frac{T}{2}} \wedge s_a = s_T \\ s_T & s_b < s_{\frac{T}{2}} \wedge s_a = s_0 \\ Neg(s_{i_1}) & s_b < s_{\frac{T}{2}} \wedge s_0 < s_a \leq s_b \\ Neg(s_{i_2}) & s_b < s_{\frac{T}{2}} \wedge s_b < s_a < s_T \\ s_0 & s_b < s_{\frac{T}{2}} \wedge s_a = s_T \end{cases}$$

tal que:

$i_1 = Max\{0, round(b - \frac{b-a}{k})\}, i_2 = Min\{T, round(b + \frac{a-b}{k})\}, k \in \{0, 1, 2, \ldots, b\}$.

g^1 se basaba en la distancia o cercanía entre el peso lingüístico del índice $Label(\mathcal{F}(d_j, d_i)) = s_a$ y el peso lingüístico del término de la consulta $c_i^1 = s_b$, siendo $Label : [0, 1] \rightarrow \mathcal{S}$ una función que asigna una etiqueta en $\mathcal{S}$ a un valor numérico $r \in [0, 1]$.

3. *Evaluación de los átomos con respecto a la semántica cuantitativa:* En este paso, los documentos son evaluados con respecto a su relevancia a átomos individuales de la consulta, pero ahora, considerando las restricciones impuestas por la semántica cuantitativa. En [51], la evaluación del átomo $\langle t_i, c_i^1, c_i^2, c_i^3 \rangle$ con respecto a la semántica cuantitativa asociada con c_i^2 para un documento d_j, llamado $RSV_j^{i,1,2} \in \mathcal{S}$, se obtenía por medio de una función de evaluación lingüística $g^2 : \mathcal{D} \times \mathcal{S}^2 \rightarrow \mathcal{S}$ como sigue:

$$RSV_j^{i,1,2} = g^2(d_j, RSV_j^{i,1}, c_i^2) \begin{cases} s_0 & d_j \notin \beta^S \\ RSV_j^{i,1} & d_j \in \beta^S \end{cases}$$

donde β^S es el conjunto de documentos tales que $\beta^S \subseteq Supp(M_i)$ donde $M_i = \{(d_1, RSV_1^{i,1}), \ldots, (d_m, RSV_m^{i,1})\}$, es un subconjunto difuso de documentos obtenidos de acuerdo a los siguientes pasos:

a) $K = \sharp Supp(M_i)$.

b) REPEAT

$M_i^K = \{s_q \in \mathcal{S} : \mu_{s_q}(\frac{K}{m}) = Sup_v\{\mu_{s_v(\frac{K}{m})}\}\}$.

$S^K = Sup_q\{s_q \in M_i^K\}$.

$K = K - 1$.

c) UNTIL$((c_i^2 \in M_i^{K+1}) \vee (c_i^2 \geq S^{K+1}))$.

d) $\beta^S = \{d_{\sigma(1)}, \ldots, d_{\sigma(K+1)}\}$, tal que $RSV_{\sigma(h)}^{i,1} \leq RSV_{\sigma(l)}^{i,1}, \forall l \leq h$.

Según g^2, la aplicación de la semántica cuantitativa consiste en reducir el número de documentos resultante de la evaluación de t_i en los últimos pasos.

4. *Evaluación de subexpresiones y modelado de la semántica de importancia relativa:* En este paso, los documentos son evaluados con respecto a su relevancia a las subexpresiones Booleanas de la consulta (combinaciones Boolean de átomos establecidos por medio de conectivos lógicos), considerando las restricciones impuestas sobre los átomos conectados por la semántica de importancia. Podemos tener dos clases de subexpresiones: conjuntivas o disyuntivas. Para modelar el conector AND este SRI usa el operador lingüístico MIN y para modelar el conector OR el operador lingüístico MAX. En el caso del conector AND, la evaluación de los pesos de importancia se introduce usando la función de transformación $MAX(Neg(weight), value)$, y en el caso del conector OR usando la función de transformación lingüística $MIN(weight, value)$.

5. *Evaluación de la consulta completa:* En este paso final de la evaluación,

los documentos son evaluados con respecto a su relevancia a combinaciones Booleanas de todas las subexpresiones existentes en la consulta. Para evaluar los conectores AND y OR usamos los operados lingüísticos MIN y MAX, respectivamente.

4.3. Un Nuevo Modelo de Sistema de Recuperación de Información Lingüístico Difuso Basado en 2-tupla

En esta Sección, presentamos un nuevo modelo de SRI lingüístico difuso basado en el enfoque lingüístico difuso 2-tupla cuya aplicación sobre la representación de la información lingüística nos permite superar los problemas detectados en [51]. La principal novedad de este nuevo modelo de SRI radica en el diseño del subsistema de evaluación que usa las ventajas del modelo lingüístico difuso 2-tupla para evitar la pérdida de información y precisión. Además, incluye un nuevo operador de soft computing, el operador LOWA 2-tupla, que se usa para modelar los conectores lógicos AND y OR en una forma más flexible.

4.3.1. Subsistema de Evaluación del Sistema de Recuperación de Información Lingüístico Difuso Basado en 2-tupla

Para definir el subsistema de evaluación asumimos que el usuario usa el mismo lenguaje de consulta presentado en Sección 4.2. Los usuarios usan consultas Booleanas lingüísticas multiponderadas para expresar sus necesidades de información cuyos pesos son valores lingüísticos ordinales. El procedimiento subyacente de este nuevo subsistema

de evaluación es similar al presentado en la Sección 4.2, esto es, la evaluación de las consultas del usuario se realiza por medio de un proceso constructivo ascendente basado en el criterio de separabilidad [91] y al mismo tiempo soportando todas las semánticas de las consultas ponderadas consideradas.

En lo que que sigue mostramos los pasos en la evaluación de este subsistema de evaluación, los cuales son definidos usando el enfoque lingüístico difuso 2-tupla.

Preprocesamiento de la consulta Como en la Sección 4.2, la consulta del usuario se preprocesa y se pone o en CNF o en DNF (véase Figura 4.1).

Evaluación de los átomos con respecto a la semántica de umbral En este paso, los documentos son evaluados de acuerdo a su relevancia solo a los átomos de la consulta, aplicando la semántica de umbral simétrico presentado en la Sección 4.2 pero definidos en un contexto lingüístico difuso 2-tupla. La función de evaluación g^1 usando el modelo de representación de información lingüística 2-tupla se llama $g^1_{2t} : \mathcal{D} \times \mathcal{T} \times \mathcal{S} \rightarrow \mathcal{S} \times [-.5, .5))$.

Entonces, dado un átomo $\langle t_i, c_i^1, c_i^2, c_i^3 \rangle$ y un documento $d_j \in \mathcal{D}$, g^1_{2t} obtiene los RSVs lingüísticos de d_j, llamados RSV_j, midiendo como de bien el peso del término índice $\mathcal{F}(d_j, t_i)$ satisface la solicitud expresada por el peso lingüístico c_i^1 según

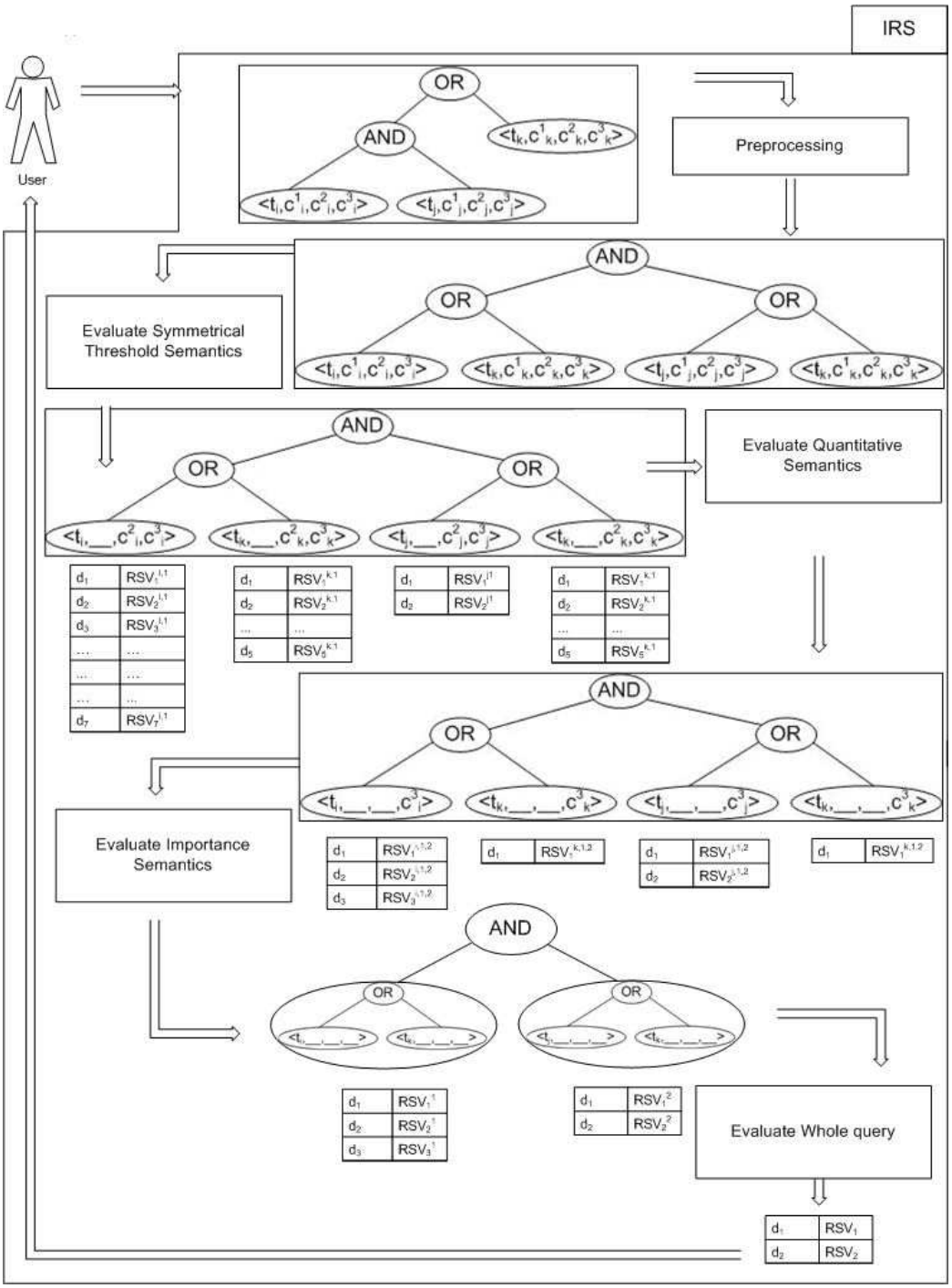

Figura 4.1: Proceso de recuperación de información detallado.

la siguiente expresión:

$$RSV_j^{i,1} = g_{2t}^1(d_j, t_i, c_i^1) = \begin{cases} (s_0,0) & (s_b,0) \geq (s_{\frac{T}{2}},0) \wedge (s_a,\alpha_a) = (s_0,0) \\ i_1 & (s_b,0) \geq (s_{\frac{T}{2}},0) \wedge (s_0,0) < (s_a,\alpha_a) < (s_b,0) \\ i_2 & (s_b,0) \geq (s_{\frac{T}{2}},0) \wedge (s_b,0) \leq (s_a,\alpha_a) < (s_T,0) \\ (s_T,0) & (s_b,0) \geq (s_{\frac{T}{2}},0) \wedge (s_a,\alpha_a) = (s_T,0) \\ (s_T,0) & (s_b,0) < (s_{\frac{T}{2}},0) \wedge (s_a,\alpha_a) = (s_0,0) \\ Neg(i_1) & (s_b,0) < (s_{\frac{T}{2}},0) \wedge (s_0,0) < (s_a,\alpha_a) \leq (s_b,0) \\ Neg(i_2) & (s_b,0) < (s_{\frac{T}{2}},0) \wedge (s_b,0) < (s_a,\alpha_a) < (s_T,0) \\ (s_0,0) & (s_b,0) < (s_{\frac{T}{2}},0) \wedge (s_a,\alpha_a) = (s_T,0) \end{cases}$$

tal que:

$i_1 = \Delta(\Delta^{-1}(s_b,0) - \frac{\Delta^{-1}(s_b,0)-\Delta^{-1}(s_a,\alpha_a)}{k})$, $i_2 = \Delta(\Delta^{-1}(s_b,0) + \frac{\Delta^{-1}(s_a,\alpha_a)-\Delta^{-1}(s_b,0)}{k})$, $k \in \{0,1,2,\ldots,b\}$, $(s_a,\alpha_a) = \Delta(T \cdot \mathcal{F}(d_j,t_i))$ y $(s_b,0)$ es el peso de umbral c_i^1 expresado en el enfoque de representación lingüístico 2-tupla.

Evaluación de los átomos con respecto a la semántica cuantitativa En este paso, los documentos son evaluados con respecto a su relevancia a átomos individuales de la consulta, pero considerando las restricciones impuestas por la semántica cuantitativa. Los pesos lingüísticos cuantitativos se interpretan como sigue [51]: cuando un usuario estable un cierto número de documentos para un término en la consulta, expresado mediante un peso lingüístico cuantitativo, entonces el conjunto de documentos a ser recuperado debe ser el mínimo número de documentos que satisfacen la compatibilidad o función de pertenencia asociada con el significado de la etiqueta usada como peso lingüístico cuantitativo. Además, aquellos documentos deben ser los que mejor satisfagan las restricciones de umbral impuestas sobre el término.

Por consiguiente, dado un átomo $\langle t_i, c_i^1, c_i^2, c_i^2\rangle$ y asumiendo que $RSV_j^{i,1} \in (\mathcal{S} \times [-.5, .5))$ representa la evaluación de acuerdo a las semántica de umbral simétrico para d_j, modelamos la interpretación de una semántica cuantitativa por medio de una función de evaluación lingüística 2-tupla, llamada g_{2t}^2. Esta función se define entre el $RSV_j^{i,1}$ y el peso lingüístico cuantitativo $c_i^2 \in \mathcal{S}^2$. El valor de evaluación del átomo $\langle t_i, c_i^1, c_i^2, c_i^3\rangle$ con respecto a c_i^2 para un documento d_j, llamado $RSV_j^{i,1,2} \in (\mathcal{S} \times [-.5, .5))$, se obtiene por medio de la función de evaluación lingüística $g_{2t}^2 : \mathcal{D} \times (\mathcal{S} \times [-.5, .5)) \times \mathcal{S} \rightarrow (\mathcal{S} \times [-.5, .5))$ definida según la siguiente expresión:

$$RSV_j^{i,1,2} = g_{2t}^2(d_j, RSV_j^{i,1}, c_i^2) = \begin{cases} (s_0, 0) & d_j \notin \beta^S \\ RSV_j^{i,1} & d_j \in \beta^S \end{cases}$$

β^S es un subconjunto de documentos obtenido según los siguientes pasos:

1. $K = \sharp Supp(M_i)$.
2. REPEAT

   $S^K = (s_e, \alpha_e) = \Delta(T \cdot \frac{K}{m})$.

   $K = K - 1$.
3. UNTIL$((s_i^2, 0) \geq S^{K+1})$
4. $\beta^S = \{d_{\sigma(1),\ldots,d_{\sigma(K+1)}}\}$, tal que $RSV_{\sigma(h)}^{i,1} \leq RSV_{\sigma(l)}^{i,1}, \forall l \leq h$.

Evaluación de las subexpresiones y modelado de la semántica de importancia relativ

Consideramos que la semántica de importancia relativa en un átomo individual no tiene sentido. Entonces, en este paso tenemos que evaluar la relevancia de los documentos con respecto a todas las subexpresiones de las consultas preprocesadas que están compuestas de un mínimo de dos componentes átomos.

Dada una subexpresión q_v con $\eta \geq 2$ átomos, sabemos que cada documento d_j presenta un $RSV_j^{i,1,2} \in (\mathcal{S} \times [-.5, .5))$ parcial con respecto a cada átomo $\langle t_i, c_i^1, c_i^2, c_i^3 \rangle$ of q_v. La evaluación de la relevancia de un documento d_j con respecto a una subexpresión completa q_v implica la agregación de los grados de relevancia parciales $\{RSV_j^{i,1,2}, i = 1, \ldots, \eta\}$ ponderados por medio de los grados de importancia relativa $\{c_i^3 \in \mathcal{S}, i = 1, \ldots, \eta\}$. Para hacer esto, necesitamos primero, definir un operador de agregación de información lingüística 2-tupla no ponderada. Este operador, que llamaremos $LOWA_{2t}$, se desarrolla a partir del operador lingüístico $LOWA$ (Definición 3.8) [44], y en base a este diseñaremos un operador de agregación de información lingüística 2-tupla ponderada, el cual debería garantizar que los términos más importantes en la consulta sean los términos más importantes en la determinación de los RSVs.

Definición 4.1. Sea $\{(a_1, \alpha_1), \ldots, (a_m, \alpha_m)\}$ un conjunto de varias 2-tupla lingüísticas a agregar, entonces el operador $LOWA_{2t}$ se define como:

$$\phi_{2t}((a_1, \alpha_1), \ldots, (a_m, \alpha_m)) = W \cdot B^T =$$

$$= C_{2t}^m\{w_w, b_w, k = 1, \ldots, m\} =$$

$$= w_1 \otimes b_1 \oplus (1 - w_1) \otimes C_{2t}^{m-1}\{\beta_h, b_h, h = 2, \ldots, m\}$$

donde $b_i = (a_i, \alpha_i) \in (\mathcal{S} \times [-.5, .5)), W = [w_1, \ldots, w_m]$, es un vector de pesos, tal que, $w_i \in [0, 1]$ and $\sum_i w_i = 1$, $\beta_h = \frac{w_h}{\sum_2^m w_k}, h = 2, \ldots, m$, y B es el vector ordenado de 2-tupla lingüísticas. Cada elemento $b_i \in B$ es el i-esimo mayor de la colección $\{(a_1, \alpha_1), \ldots, (a_m, \alpha_m)\}$, y C_{2t}^m es el operador de combinación convexa de m 2-tupla. Si $w_j = 1$ y $w_i = 0$ con $i \neq j \forall i, j$ la combinación convexa se define

como: $C_{2t}^{m}\{w_i, b_i, i = 1, \ldots, m\} = b_j$. Y si $m = 2$ entonces se define como:

$$C_{2t}^{2}\{w_l, b_l, l = 1, 2\} = w_1 \otimes b_j \oplus (1 - w_1) \otimes b_i = \tau_t^{t'}(s_k^{n(t')}, \alpha)$$

donde $(s_k^{n(t')}, \alpha) = \Delta(\lambda)$ y $\lambda = \Delta^{-1}(\tau_{t'}^{t}(b_i)) + w_1 \cdot (\Delta^{-1}(\tau_{t'}^{t}(b_j)) - \Delta^{-1}(\tau_{t'}^{t}(b_i)))$, $b_j, b_i \in (\mathcal{S} \times [-.5, .5))$, $(b_j \geq b_i)$, $\lambda \in [0, n(t') - 1], t \in \{t^-, t^+\}$.

También podemos definir un operador ponderado de información lingüística 2-tupla.

Generalmente, un operador ponderado de agregación de information realiza dos actividades [42]: i) la transformación de la información ponderada bajo los grados ponderados, y ii) la agregación de la información ponderada transformada por medio de un operador de agregación de información no ponderada.

1. La transformación de la información ponderada bajo los grados ponderados por medio de la función de transformación h. Algunos familias de conectivos usados como funciones de transformación en un entorno no balanceado son los siguientes dos:

   a) *Funciones de conjunción lingüística ($LC^{\rightarrow}$).* Las funciones conjuntivas que usaremos son las siguientes t-normas, las cuales son monotonamente no decrecientes en los pesos y satisfacen las propiedades requeridas por cualquier función de transformación, h, [36]:

   - Clásico operador MIN:

   $$LC_1^{\rightarrow}(\omega, a) = MIN_{2t}(\omega, a),$$

   donde MIN_{2t} es el operador de comparación 2-tupla.

- Nilpotent MIN:

$$LC_2^{\rightarrow}(\omega, a) = \begin{cases} MIN_{2t}(\omega, a) & \text{if } \omega > NEG(a) \\ s_0 & \text{en otro caso.} \end{cases}$$

- Conjunción débil:

$$LC_3^{\rightarrow}(\omega, a) = \begin{cases} MIN_{2t}(\omega, a) & \text{if } MAX_{2t}(\omega, a) = s_T \\ s_0 & \text{en otro caso.} \end{cases}$$

donde $\omega, a \in \mathcal{S} \times [-.5, .5)$, a es parte de la información a agregar y ω es el peso asociado a a.

b) *Funciones de implicación lingüística ($LI^{\rightarrow}$).* Las funciones de implicación lingüística que usaremos son monotonamente no crecientes en los pesos y satisfacen las propiedades requeridas para cualquier función de transformación h [36]:

- Función de implicación de Kleene-Dienes:

$$LI_1^{\rightarrow}(\omega, a) = MAX_{2t}(NEG(\omega), a),$$

donde MAX_{2t} es el operador de comparación 2-tupla.

- La función de implicación de Gödel:

$$LI_2^{\rightarrow}(\omega, a) = \begin{cases} s_T & \text{if } \omega \leq a \\ a & \text{en otro caso.} \end{cases}$$

- Función de implicación de Fodor:

$$LI_3^{\rightarrow}(\omega, a) = \begin{cases} s_T & \text{if } \omega \leq a \\ MAX_{2t}(NEG(\omega), a) & \text{en otro caso.} \end{cases}$$

donde $\omega, a \in \mathcal{S} \times [-.5, .5)$, a es parte de la información lingüística a agregar y ω es el peso asociado a a.

2. La agregación de la información ponderada transformada por medio de un operador de agregación de información no ponderada f. Como es sabido, la elección de h depende de la de f.

   Como operador f podemos usar $LOWA_{2t}$.

En [92], Yager discutió el efecto de los grados de importancia sobre las agregaciones de tipo MAX y MIN y sugirió una clase de funciones para transformaciones de importancia en ambos tipos de agregación. Para la agregación MIN, sugirió una familia de t-conormas actuando sobre la información ponderada y la negación del grado de importancia, lo cual presenta la propiedad de ser monotomamente no creciente en estos grados de importancia. Para la agregación tipo MAX, sugirió una familia de t-normas actuando sobre la información ponderada y el grado de importancia, lo cual presenta la propiedad de ser monotonamente no decreciente en estos grados de importancia.

Siguiendo las recomendaciones de Yager, en [51] se propuso modelar las subexpresiones conjuntivas por medio de la t-norma lingüística MIN y la transformación de la información ponderada bajo los grados de importancia por medio de la función de implicación lingüística MAX(NEG(weight),valor), y las subexpresiones disyuntivas por medio de la t-conorma lingüística MAX y la transformación de la información ponderada bajo los grados de importancia por medio de la t-norma lingüística MIN.

Sin embargo, como es sabido, la evaluación de los conectores lógicos AND y OR por medio de los operadores MIN y MAX presenta algunas limitaciones. Esto es, puede causar un muy restrictivo e inclusivo comportamiento, respectivamente. El

problema es que el proceso de recuperación puede ser engañoso ya que, por un lado, la t-norma lingüística MIN puede causar el rechazo de documentos útiles por la no satisfacción de algún criterio en alguna de las subexpresiones conjuntivas y, por otro lado, la t-conorma lingüística MAX puede provocar la aceptación de documentos útiles por la simple satisfacción de algún criterio en sólo alguna de las subexpresiones disyuntivas.

Por tanto, para evaluar las subexpresiones junto con la semántica de importancia relativa y de acuerdo a las actividades necesarias para agregar información ponderada, si una subexpresión es conjuntiva usamos $f = \phi_{2t}^1$ y $h = MAX_{2t}(NEG(weight, 0), 2-tuple\_value)$, y si es disyuntiva usamos $f = \phi_{2t}^2$, con $h = MIN_{2t}((weight, 0), 2 - tuple\_value)$, siendo MAX_{2t} y MIN_{2t} operadores de comparación de 2-tupla y ϕ_{2t}^1 es el operador ϕ_{2t} con un $orness(W) \leq 0.5$ y ϕ_{2t}^2 es el operador ϕ_{2t} con un $orness(W) > 0.5$.

De manera resumida, dado un documento d_j, evaluamos su relevancia con respecto a una subexpresión q_v, llamado $RSV_j^v \in (\mathcal{S} \times [-.5, .5))$ como:

1. Si q_v es una subexpresión conjuntiva entonces

$$RSV_j^v = \phi_{2t}^1(MAX_{2t}(Neg(c_1^3, 0), RSV_j^{1,1,2}), \ldots, MAX_{2t}(Neg(c_\eta^3, 0), RSV_j^{\eta,1,2})).$$

2. Si q_v es una subexpresión disyuntiva entonces

$$RSV_j^v = \phi_{2t}^2(MIN_{2t}((c_1^3, 0), RSV_j^{1,1,2}), \ldots, MIN_{2t}((c_\eta^3, 0), RSV_j^{\eta,1,2})).$$

Evaluación de la consulta completa En este paso, la evaluación final de cada documento se lleva combinando sus evaluaciones con respecto a todas las subexpre-

siones. Para hacer esto, usamos de nuevo ambos operadores LOWA 2-tupla ϕ_{2t}^1 y ϕ_{2t}^2 para modelar los conectores AND y OR, respectivamente.

Por tanto, dado un documento d_j, su relevancia con respecto a una consulta, $RSV_j \in (\mathcal{S} \times [-.5, .5))$, se obtiene como:

1. Si q está en CNF entonces $RSV_j = \phi_{2t}^1(RSV_j^1, \ldots, RSV_j^v)$, y
2. Si q está en DNF entonces $RSV_j = \phi_{2t}^2(RSV_j^1, \ldots, RSV_j^v)$,

siendo v el número de subexpresiones de q.

Este proceso de evaluación se muestra en la Figura 4.1.

NOTA: Sobre el operador NOT. Deberíamos notar que, si una consulta está en CNF o DNF, tenemos que definir el operador de negación solo a nivel de átomos simples. Esto simplifica la definición del operador NOT. Como se hizo en [51], la evaluación del documento d_j para un átomo ponderado negado $\langle \neg t_i, c_i^1, c_i^2, c_i^3 \rangle$ se obtiene de la negación del peso del término índice $\mathcal{F}(t_i, d_j)$. Esto significa calcular la función de evaluación de la semántica de umbral g_{2t}^1 del valor lingüístico 2-tupla $(s_a, \alpha_a) = \Delta(T \cdot (1 - \mathcal{F}(d_j, t_i)))$.

Resumidamente, este subsistema de evaluación puede ser sintetizado por medio de una función de evaluación general $\mathcal{E}_{2t} : \mathcal{D} \times \mathcal{Q} \rightarrow (\mathcal{S} \times [-.5, .5))$, que evalúa las diferentes clases de consultas preprocesadas, $\{q = \langle t_i, c_i^1, c_i^2, c_i^3 \rangle, q \wedge p, q \vee p, \neg q\}$ de acuerdo a las siguientes cinco reglas:

1. Átomos:

$$\mathcal{E}_{2t}(d_j, q^1) = g^2_{2t}(d_j, g^1_{2t}(d_j, t_i, c^1_i), c^2_i),$$

tal que $q^1 = \langle t_i, c^1_i, c^2_i, c^3_i \rangle$.

2. Subexpresiones Conjuntivas:

$$\mathcal{E}_{2t}(d_j, q^2) = \phi^1_{2t}(MAX_{2t}(Neg(c^3_1, 0), \mathcal{E}_{2t}(d_j, q^1_1)),$$

$$\ldots, MAX_{2t}(Neg(c^3_\eta, 0), \mathcal{E}_{2t}(d_j, q^1_\eta))),$$

siendo η el número de átomos de q^2.

3. Subexpresiones Disyuntivas:

$$\mathcal{E}_{2t}(d_j, q^3) = \phi^2_{2t}(MIN_{2t}((c^3_1, 0), \mathcal{E}_{2t}(d_j, q^1_1)),$$

$$\ldots, MIN_{2t}((c^3_\eta, 0), \mathcal{E}_{2t}(d_j, q^1_\eta))).$$

4. Consulta en CNF:

$$\mathcal{E}_{2t}(d_j, q^4) = \phi^1_{2t}(\mathcal{E}_{2t}(d_j, q^3_1), \ldots, \mathcal{E}_{2t}(d_j, q^3_\omega))$$

siendo ω el número de subexpresiones disyuntivas.

5. Consulta en DNF:

$$\mathcal{E}_{2t}(d_j, q^5) = \phi^2_{2t}(\mathcal{E}_{2t}(d_j, q^2_1), \ldots, \mathcal{E}_{2t}(d_j, q^2_\omega))$$

siendo ω el número de subexpresiones disyuntivas.

Entonces, el resultado del sistema para cualquier consulta q es un subconjunto de documentos caracterizados por una función de pertenencia lingüística $\mathcal{E}_{2t}$:

$$\{(d_1, \mathcal{E}_{2t}(d_1, q^k)), \ldots, (d_m, \mathcal{E}_{2t}(d_m, q^k))\}, k \in 1, 2, 3, 4, 5.$$

Los documentos se muestran en orden decreciente de $\mathcal{E}_{2t}$ y asociados en clases de relevancia lingüística, de tal manera que el número máximo de clases está limitado por la cardinalidad del conjunto de etiquetas elegidas para representar la variable lingüística *Relevancia*.

4.3.2. Ejemplo Teórico del Rendimiento del Nuevo Sistema de Recuperación de Información Lingüístico 2-tupla Definido

En esta subsección, presentamos un ejemplo de rendimiento del SRI propuesto y comparamos su rendimiento con aquel definido en [51].

Supongamos una pequeña base de datos con un conjunto de siete documentos $\mathcal{D} = \{d_1, \ldots, d_7\}$, representado por medio de un conjunto de diez términos índice $\mathcal{T} = \{t_1, \ldots, t_{10}\}$. Los documentos son indexados por medio de una función de indexación numérica $\mathcal{F}$, que representa a estos documentos como sigue:

$d_1 = 0.7/t_5 + 0.4/t_6 + 1/t_7$

$d_2 = 1/t_4 + 0.6/t_5 + 0.8/t_6 + 0.9/t_7$

$d_3 = 0.5/t_2 + 1/t_3 + 0.8/t_4$

$d_4 = 0.9/t_4 + 0.5/t_6 + 1/t_7$

$d_5 = 0.7/t_3 + 1/t_4 + 0.4/t_5 + 0.8/t_9 + 0.6/t_10$

$d_6 = 0.8/t_5 + 0.99/t_6 + 0.8/t_7$

$d_7 = 0.8/t_5 + 0.02/t_6 + 0.8/t_7 + 0.9/t_8$

Asumiendo el siguiente conjunto de nueve etiquetas $\mathcal{S} = \{N, EL, VL, L, M, H, VH, EH, T\}$ representamos estos documentos usando el enfoque lingüístico difuso 2-tupla aplicando la función Δ sobre los pesos del término índice $\mathcal{F}(d_j, t_i)$:

$d_1 = (VH, -.4)/t_5 + (L, .2)/t_6 + (T, 0)/t_7$

$d_2 = (T, 0)/t_4 + (H, -.2)/t_5 + (VH, .4)/t_6 + (EH, .2)/t_7$

$d_3 = (M, 0)/t_2 + (T, 0)/t_3 + (VH, .4)/t_4$

$d_4 = (EH, .2)/t_4 + (M, 0)/t_6 + (T, 0)/t_7$

$d_5 = (VH, -.4)/t_3 + (T, 0)/t_4 + (L, .2)/t_5 + (VH, .4)/t_9 + (H, -.2)/t_10$

$d_6 = (VH, .4)/t_5 + (T, -.08)/t_6 + (VH, .4)/t_7$

$d_7 = (VH, .4)/t_5 + (N, .16)/t_6 + (VH, .4)/t_7 + (EH, .2)/t_8$

Suponemos que un usuario formula los siguiente consulta ponderada lingüística:

$$q = ((t_5, VH, VL, VH) \wedge (t_6, L, L, VL)) \vee (t_7, H, L, H).$$

Preprocesamiento de la consulta La consulta q está en DNF, pero presenta una subexpresión con un solo átomo. Por tanto, q debe ser preprocesada y transformada en forma normal con todas las subexpresiones con un mínimo de dos átomos. Entonces, q es transformada en la siguiente consulta equivalente:

$$q^{'} = ((t_5, VH, VL, VH) \vee (t_7, H, L, H)) \wedge ((t_6, L, L, VL) \vee (t_7, H, L, H)),$$

que está expresada en CNF.

Evaluación de los átomos con respecto a la semántica de umbral simétrico

Después que q se transforme en forma normal, evaluamos sus átomos según la semántica de umbral simétrico por medio de g^1_{2t} y obtenemos lo que sigue:

- $\{RSV_1^{5,1} = (VH, -.2), RSV_2^{5,1} = (H, .4), RSV_5^{5,1} = (H, -.4),$
  $RSV_6^{5,1} = (VH, .2), RSV_7^{5,1} = (VH, .2)\}$
- $\{RSV_1^{6,1} = (H, -.1), RSV_2^{6,1} = (L, .3), RSV_4^{6,1} = (H, -.5),$
  $RSV_6^{6,1} = (L, -.46), RSV_7^{6,1} = (VH, .42)\}$

- $\{RSV_1^{7,1} = (T, 0), RSV_2^{7,1} = (VH, .1), RSV_4^{7,1} = (T, 0),$
  $RSV_6^{7,1} = (VH, -.3), RSV_7^{7,1} = (VH, -.3)\}$

donde, por ejemplo $RSV_2^{7,1}$ se calcula como sigue:

$$RSV_2^{7,1} = g_{2t}^1(d_2, t_7, H) = i_2 =$$

$$= \Delta(\Delta^{-1}(H, 0) + \frac{\Delta^{-1}(EH, .2) - \Delta^{-1}(H, 0)}{2}) = (VH, .1),$$

(con $k = 2$), dado que la condición $(s_b, 0) \geq (s_{\frac{T}{2}}) \wedge (s_b, 0) \leq (s_a, \alpha_a) < (s_T, 0)$ es cierta.

Si aplicamos el modelo SRI lingüístico ordinal [51], esto es, la función de evaluación g^1, obtenemos los grados lingüísticos de relevancia siguientes:

- $\{RSV_1^{5,1} = VH,\ RSV_2^{5,1} = H,\ RSV_5^{5,1} = H,\ RSV_6^{5,1} = VH,\ RSV_7^{5,1} = VH\}$
- $\{RSV_1^{6,1} = H,\ RSV_2^{6,1} = L,\ RSV_4^{6,1} = H,\ RSV_6^{6,1} = L,\ RSV_7^{6,1} = VH\}$
- $\{RSV_1^{7,1} = T,\ RSV_2^{7,1} = VH,\ RSV_4^{7,1} = T,\ RSV_6^{7,1} = VH,\ RSV_7^{7,1} = VH\}$

En este paso, es fácil observar los efectos del uso de la representación lingüística 2-tupla. Obviamente, los resultados lingüísticos parciales con el modelo 2-tupla son muchos más ricos que con el modelo lingüístico ordinal.

Evaluación de los átomos con respecto a la semántica cuantitativa La evaluación de los átomos de q de acuerdo a la semántica cuantitativa modelada por g_{2t}^2 son:

- $\{RSV_6^{5,1,2} = (VH, .2)\}$
- $\{RSV_1^{6,1,2} = (H, -.1), RSV_7^{6,1,2} = (VH, .42)\}$

- $\{RSV_1^{7,1,2} = (T, 0), RSV_4^{7,1,2} = (T, 0)\}$

donde, por ejemplo, $RSV_1^{7,1,2} = g_{2t}^2(d_2, RSV_1^{7,1}, c_7^2)$ se calcula como:

$K = \sharp Supp(M_7) = 5$, dado que $Supp(M_7) = \{d_1, d_2, d_4, d_6, d_7\}$, cuando $K = 2$ la condición $(c_7^2, 0) = (L, 0) \geq (VL, .28) = S^2$ es cierta por tanto, obtenemos $\beta^S = \{d_1, d_4\}$, así, $RSV_1^{7,1,2} = g_{2t}^2(d_2, RSV_1^{7,1}, c_7^2) = RSV_1^{7,1} = (T, 0)$, ya que $d_1 \in \beta^S$.

Los resultados obtenidos por g^2 definida en [51] son:

- $\{RSV_6^{5,1,2} = VH\}$
- $\{RSV_1^{6,1,2} = H, RSV_7^{6,1,2} = VH\}$
- $\{RSV_1^{7,1,2} = T, RSV_4^{7,1,2} = T\}$

Deberíamos notar que la semántica cuantitativa decrementa el número de documentos a cada término de la consulta. En realidad la representación lingüística 2-tupla no afecta a este paso de la evaluación.

Evaluación de subexpresiones y modelado de la semántica de importancia relativa

La consulta q' tiene dos subexpresiones y ambas presentan dos átomos, $q_1' = (t_5, VH, VL, VH) \vee (t_7, H, L, H)$ y $q_2' = (t_6, L, L, VL) \vee (t_7, H, L, H)$. Cada subexpresión está en forma disyuntiva, y por tanto, debemos usar un operador LOWA 2-tupla ϕ_{2t}^2 con una medida $orness(W) > 0.5$ (por ejemplo, con $(W = [0.8, 0.2])$) junto con la función transformación lingüística $MIN_{2t}(Weight, 2 - tuple\_value)$ para evaluarlas. Por tanto, los resultados de la evaluación después de aplicar la semántica de importancia relativa son:

- $\{RSV_1^1 = (M, 0), RSV_4^1 = (M, 0), RSV_6^1 = (H, -.2)\}$

- $\{RSV_1^2 = (M, .4), RSV_4^2 = (M, 0), RSV_7^2 = (VL, -.4)\}$

donde RSV_j^v es el resultado de evaluar el documento d_j con respecto a la subexpresión $q_v^{'}, v \in \{1, 2\}$.

Por ejemplo RSV_1^2 se calcula como sigue:

$$RSV_1^2 = \phi_{2t}^2(MIN_{2t}((c_6^3, 0), RSV_1^{6,1,2}), MIN_{2t}((c_7^3, 0), RSV_1^{7,1,2}))$$

Esto es,

$$RSV_1^2 = \phi_{2t}^2(MIN_{2t}((VL, 0), (H, -.1)), MIN_{2t}((H, 0), (T, 0))) =$$

$$= \phi_{2t}^2((VL, 0), (H, 0)) \Rightarrow^{\sigma}$$

$$RSV_1^2 = \phi_{2t}^2((H, 0), (VL, 0)) = \Delta(\Delta^{-1}(H, 0) \cdot 0.8 + \Delta^{-1}(VL, 0) \cdot 0.2) =$$

$$= \Delta(5 \cdot 0.8 + 2 \cdot 0.2) = \Delta(4.4) = (M, .4).$$

Usando la t-conorma lingüística MAX junto con la función de transformación $MIN_{2t}(Weight, value)$ para las subexpresiones disyuntivas obtenemos lo siguiente:

- $\{RSV_1^1 = H, RSV_4^1 = H, RSV_6^1 = VH\}$
- $\{RSV_1^2 = H, RSV_4^2 = H, RSV_7^2 = VL\}$.

Deberíamos apuntar que en general el operador lingüístico LOWA 2-tupla disminuye el efecto inclusivo de la t-conorma lingüística MAX para calcular los grados de relevancia lingüísticos.

Evaluación de la consulta completa Obtenemos la evaluación de la consulta completa usando un operador lingüístico LOWA 2-tupla ϕ_{2t}^1 con $orness(W) < 0.5$ (por ejemplo con $(W = [0.2, 0.8])$).

$$\{RSV_1 = (M, .08), RSV_4 = (M, 0), RSV_6 = (EL, -.08), RSV_7 = (N, .32)\}.$$

El mejor documento recuperado es d_1, el cual se calcula como:

$$RSV_1 = \phi_{2t}^1(RSV_1^2, RSV_1^1) = \phi_{2t}^1((M, .4), (M, 0)) =$$

$$= \Delta(\Delta^{-1}(M, .4) \cdot 0.2 + \Delta^{-1}(M, 0) \cdot 0.8) = \Delta(4.08) = (M, .08).$$

El resultado obtenido si usáramos la t-norma MIN sería:

$$\{RSV_1 = H, RSV_4 = H\}.$$

En este caso, se obtienen los dos mejores documentos, d_1 and d_4, sin posibilidad de distinguir entre ellos.

En la Figura 4.2 mostramos gráficamente el ejemplo completo de rendimiento de este nuevo modelo SRI lingüístico.

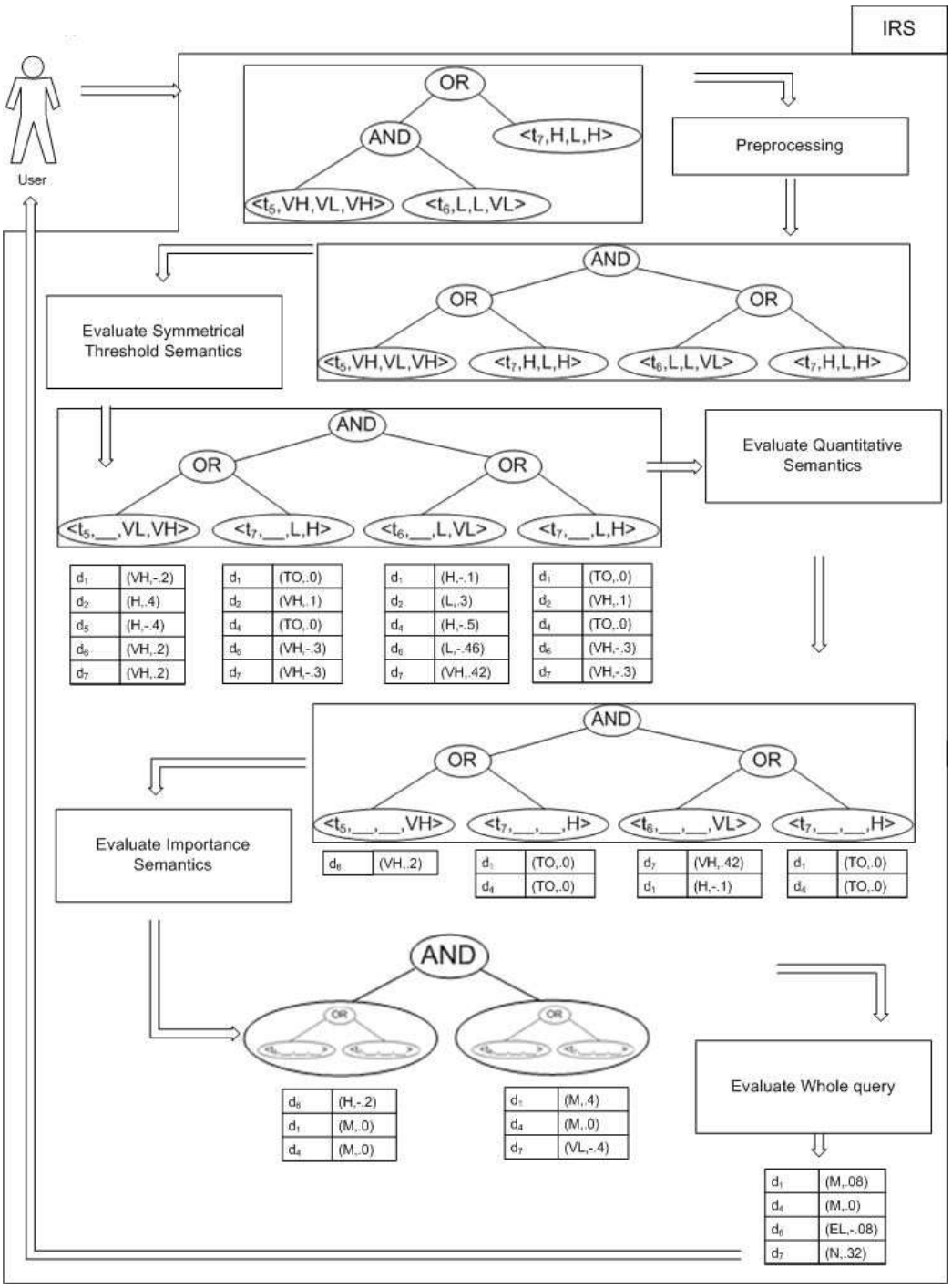

Figura 4.2: Ejemplo de proceso de recuperación de información.

4.3.3. Ejemplo Práctico del Rendimiento del Nuevo Sistema de Recuperación de Información Lingüístico 2-tupla Definido

En este apartado mostramos un ejemplo práctico de rendimiento del nuevo modelo de SRI propuesto en Sección 4.3, al que llamaremos SRI_{2t}. Los resultados de SRI_{2t} serán comparados con los obtenidos por el modelo de SRI ordinal propuesto en [51] y descrito en la Sección 4.2. A este último lo denotaremos por SRI_o.

Para mostrar el benificio de usar el modelo de representación de información lingüística 2-tupla, junto con el nuevo operador de agregación $LOWA_{2t}$, en el diseño de SRI_{2t}, usaremos dos pequeños ejemplos prácticos. Éstos consisten en dos pequeñas consultas evaluadas sobre la colección TREC. Tanto las consultas como sus resultados pueden verse en las Tablas 4.1 a 4.4.

Para más ejemplos véase el Anexo B, en el cual se podrán encontrar también los documentos en los que aparecen los términos usados en los siguientes ejemplos.

Como puede verse, todas las bondades de SRI_{2t} mostradas en el ejemplo teórico del apartado anterior, son extrapolables al entorno práctico.

En primer lugar, SRI_{2t}, al no perder información es capaz de obtener un mejor ranking de documentos (Tabla 4.2), que el obtenido por SRI_o (Tabla 4.1).

En segundo lugar, el uso de operadores de agregación más flexibles que los clásicos MIN y MAX nos permite amortiguar el comportamiento extremo (es decir, rechazamos por completo un documento si no aparece en todos los operandos que intervienen; o aceptamos cualquier documento con tal de que sea recuperado por alguno de los

	RESULTADOS DE LA CONSULTA	
	Parametros utilizados	
Jerarquia = (3,3)	*Base de Datos* = "trec"(5000 Docs.)	*orness* = _
	Numero de documentos recuperados = 13	
Rank	**ID Doc**	**RSV**
1#	4984	EL
2#	4782	EL
3#	4459	EL
4#	4220	EL
5#	4157	EL
6#	4133	EL
7#	4097	EL
8#	2621	EL
9#	2423	EL
10#	1980	EL
11#	1816	EL
12#	185	EL
13#	3030	EL

Tabla 4.1: Evaluación de $\langle clamp, H, VL, \_\rangle$ con RSV_o.

	RESULTADOS DE LA CONSULTA	
	Parametros utilizados	
Jerarquia = (3,3)	*Base de Datos* = "trec"(5000 Docs.)	*orness* = _
	Numero de documentos recuperados = 13	
Rank	**ID Doc**	**RSV**
1#	4220	(L,0.13)
2#	3030	(L,0.13)
3#	4133	(L,0.01)
4#	4782	(L,-0.08)
5#	4157	(L,-0.12)
6#	4459	(L,-0.13)
7#	2621	(L,-0.15)
8#	4097	(L,-0.17)
9#	4984	(L,-0.18)
10#	185	(L,-0.18)
11#	1816	(L,-0.22)
12#	2423	(L,-0.23)
13#	1980	(L,-0.24)

Tabla 4.2: Evaluación de $\langle clamp, H, VL, \_\rangle$ con RSV_{2t}.

	RESULTADOS DE LA CONSULTA	
	Parametros utilizados	
Jerarquia = (3,3)	*Base de Datos* = "trec"(5000 Docs.)	*orness* = 1.00
	Numero de documentos recuperados = 0	
Rank	**ID Doc**	**RSV**

Tabla 4.3: Evaluación de $\langle bay, H, VL, \_\rangle AND \langle clamp, T, EL, \_\rangle$ con SRI_o.

	RESULTADOS DE LA CONSULTA	
	Parametros utilizados	
Jerarquia = (3,3)	*Base de Datos* = "trec"(5000 Docs.)	*orness* = 0.80
	Numero de documentos recuperados = 2	
Rank	**ID Doc**	**RSV**
1#	185	(L,-0.25)
2#	2423	(L,-0.28)

Tabla 4.4: Evaluación de $\langle bay, H, VL, \_\rangle AND \langle clamp, T, EL, \_\rangle$ con SRI_{2t}.

términos que intervienen en la agregación) de los conectivos lógicos AND y OR, respectivamente. Estos resultados pueden verse en las Tablas 4.3, para los resultados obtenidos por SRI_o, y 4.4 para los obtenidos por SRI_{2t}.

4.3.4. Ventajas y Desventajas

En esta subsección, analizamos las principales ventajas y desventajas de nuestra propuesta con respecto al modelo de SRI lingüístico ordinal definido en [51].

Ventajas. Observamos las siguientes ventajas:

- En primer lugar, es obvia la ventaja del uso del modelo de representación lingüístico difuso 2-tupla, dado que si usamos una representación lingüística ordinal sería imposible distinguir entre d_1 and d_4.
- Segundo, también es obvio que el modelo de representación de información lingüístico difuso 2-tupla evita la pérdida de información en el proceso de computación de los grados de relevancia.
- Tercero, con el modelo de representación lingüístico difuso 2-tupla se simplifica la complejidad de algunas funciones de evaluación, como es el caso de la semántica cuantitativa.
- En cuarto lugar, el nuevo operador propuesto para modelar los conectores lógicos AND y OR, el operador lingüístico LOWA 2-tupla, mejora el cálculo de los grados de relevancia.
- Finalmente, deberíamos apuntar que este nuevo modelo de SRI lingüístico mejora en general el rendimiento de aquel propuesto en [51] con un mínimo coste y sin afectar negativamente a la interacción usuario-SRI, dado que el lenguaje de consulta es el mismo y los grados de relevancia siguen siendo expresados en una forma lingüística.

Desventajas. Observamos desventajas similares a las que afectan al modelo de SRI propuesto en [51]. Principalmente dos:

- Con el subsistema de consulta el usuario puede expresar una gran cantidad de restricciones, pero debe decidir qué y cuántas semánticas va a considerar en la formulación de sus necesidades de información, el subsistema soporta todas las posibilidades. Por tanto, es necesario diseñar una adecuada interfaz de usuario que pueda ayudar al usuario a hacer un mejor uso de las posibilidades de expresión del lenguaje de consulta ponderado.

- Definir herramientas que permitan al usuario controlar la agregación en el proceso de evaluación, es decir, implicar el concepto de relevancia de usuario en nivel de los conectores lógicos Booleanos.

- g_{2t}^1 tiende a sobrevalorar la satisfacción/no-satisfacción de las respuestas. Este problema es una consecuencia de la propia definición de g_{2t}^1 como veremos en la Sección 4.4. En resumen podríamos decir que la evaluación es demasiado optimista cuando se satisface el valor de umbral, además de que se reducen las posibilidades de discriminación entre documentos que satisfacen el umbral. Igualmente ocurre para el caso de no-satisfacción. En la siguiente sección propondremos una mejora al nuevo modelo de SRI lingüístico propuesto en este capítulo para tratar de solucionar este último problema.

4.4. Mejoras Adicionales. Una Nueva Función de Evaluación basada en 2-tupla para Modelar la Semántica de Umbral Simétrico

Según la semántica de umbral simétrico, el subsistema de evaluación asume que un usuario puede buscar documentos con una aceptable mínima presencia de un término

en su representación (como ocurre en la interpretación clásica del umbral [61]), o documentos con una presencia máxima aceptable de un término en su representación. Por tanto, cuando un usuario solicita documentos en los cuales el concepto representado por el término t_i esté con el valor *Alto* de Importancia, el usuario no rechazará un documento con un valor $\mathcal{F}$ por encima de *Alto*; o por el contrario, cuando un usuario busca documentos en los cuales el concepto representado por t_i tenga un valor *Bajo* de Importancia, dicho usuario no rechazará documentos donde el valor $\mathcal{F}$ sea inferior a *Bajo*.

Por tanto, las consultas del tipo: $\langle t_i, c_i \rangle \in \mathcal{T} \times \mathcal{S}$, donde $c_i \geq s_{\frac{T}{2}}$ (por ejemplo: *Alto, Muy Alto, ...*), indicando la presencia del t_i en el documento, serán tratadas de manera distinta a las consultas donde $c_i < s_{\frac{T}{2}}$ (por ejemplo: *Muy Bajo, Bajo, ...*), que expresan la ausencia del término en el documento. En el primer caso, la consulta $\langle t_i, c_i \rangle$ es equivalente a $\langle t_i$, al menos $c_i \rangle$, que indica que los documentos deseados son aquellos que tienen un valor $\mathcal{F}$ tan alto como sea posible; por el contrario, la segunda situación es equivalente a $\langle t_i$, como mucho $c_i \rangle$ donde ahora, los documentos más relevantes son aquellos donde $\mathcal{F}$ es tan bajo como sea posible.

La función de evaluación g^1 definida en [51] representa una posible modelización del significado de la semántica de umbral simétrico. Otra posible interpretación para esta semántica fue presentada en [50], a la que llamaremos $g^{1'}$ y que se define como $g^{1'} : \mathcal{D} \times \mathcal{T} \times \mathcal{S} \longrightarrow \mathcal{S}$:

$$RSV_j^{i,1} = g^{1'}(d_j, t_i, c_i^1) = \begin{cases} s_{MIN\{a+\beta,T\}} & s_{\frac{T}{2}} \leq s_b \leq s_a \\ s_{MAX\{0,a-\beta\}} & (s_{\frac{T}{2}} \leq s_b) \wedge (s_a < s_b) \\ NEG(s_{MAX\{0,a-\beta\}}) & s_a \leq s_b < s_{\frac{T}{2}} \\ NEG(s_{MIN\{a+\beta,T\}}) & (s_b < s_{\frac{T}{2}}) \wedge (s_b < s_a) \end{cases}$$

tal que, (i) $s_b = c_i^1$; (ii) s_a es el peso lingüístico del término índice obtenido como $s_a = Label(\mathcal{F}(d_j, t_i))$, siendo $Label : [0,1] \longrightarrow \mathcal{S}$ una función que asigna una etiqueta en $\mathcal{S}$ al valor numérico $r \in [0,1]$; y (iii) $\beta = round(\frac{2|b-a|}{T})$ es un valor de bonificación que recompensa o penaliza los grados de relevancia según la satisfacción/no-satisfacción de la consulta $\langle t_i, c_i^1, c_i^2, c_i^3 \rangle$.

Ambas interpretaciones tienen un comportamiento similar y presentan los mismos problemas:

Pérdida de Precisión Este problema es una cosecuencia del marco lingüístico ordinal en el que se basan, que trabaja con dominios de expresión lingüísticos discretos y esto implica asumir limitaciones en el dominio de representación de los RSVs. Por consiguiente, como el conjunto de términos $\mathcal{S}$ tiene una cardinalidad limitada (5, 7 o 9 etiquetas) para establecer los RSVs lingüísticos, en consecuencia, es difícil distinguir o especificar qué documentos satisfacen realmente mejor las consultas atómicas $\langle t_i, c_i^1, c_i^2, c_i^3 \rangle$. Aunque el sistema recupere muchos documentos, los posibles juicios de relevancia están limitados por la cardinalidad del conjunto de etiquetas considerado.

Pérdida de Información Este problema es también una consecuencia del enfoque lingüístico ordinal ya que esto nos fuerza a aplicar operaciones de aproximación

en la definición de g^1 y $g^{1'}$, en particular, la operación *round*, y como se sabe [46] esto siempre causa que exista pérdida de información.

Example 4.1. Sea $\mathcal{S} = \{s_0 = \text{Null (N)}, s_1 = \text{Extremely Low (EL)}, s_2 = \text{Very Low (VL)}, s_3 = \text{Low (L)}, s_4 = \text{Medium (M)}, s_5 = \text{High (H)}, s_6 = \text{Very High (VH)}, s_7 = \text{Extremely High (EH)}, s_8 = \text{Total (T)}\}$ un conjunto de etiquetas usadas para asociar la información lingüística en un SRI y consideremos dos documentos d_1 y d_2, tales que $Label(\mathcal{F}(d_1, t_i)) = EH$ y $Label(\mathcal{F}(d_2, t_i)) = T$, respectivamente, entonces si tenemos una consulta atómica $\langle t_i, M, \_, \_\rangle$ obtenemos el mismo grado de relevancia para ambos documentos como consecuencia de la pérdida de información, $g(d_1, t_i, M) = T$ y $g^{1'}(d_2, t_i, M) = T$.

g^1 y $g^{1'}$ tienden a sobrevalorar la satisfacción/no-satisfacción de las respuestas

Este problema es una consecuencia de la propia definición de g^1 y $g^{1'}$. Por ejemplo, si analizamos su definición podemos observar que los grados de relevancia generados cuando se satisface el valor de umbral, es decir, $s_{MIN\{a+\beta,T\}}$, siempre están limitados por el peso del término índice s_a. En resumen podemos decir que la evaluación es demasiado optimista cuando se satisface el valor de umbral, además de que se reducen las posibilidades de discriminación entre documentos que satisfacen el umbral. Igualmente ocurre para el caso de no-satisfacción.

En esta sección, presentamos una nueva función de evaluación para modelar la semántica de umbral simétrico que supera los problemas, anteriormente comentados, de la funciones g^1 [51] y $g^{1'}$ [50].

La nueva función de evaluación, que llamamos $g_{2t}^{1'}$, se ha diseñado usando como base el modelo de representación de información lingüística 2-tupla [46].

Primeramente, hay que destacar que el simple hecho de definir la nueva función $g_{2t}^{1'}$ con el enfoque lingüístico 2-tupla, nos permite solventar el primer problema de g^1 y $g^{1'}$, dado que $g_{2t}^{1'}$ heredará las propiedades del modelo 2-tupla, como sucede con la función g_{2t}^1. La principal propiedad de este modelo es eliminar la pérdida de precisión del modelo lingüístico ordinal [46]. Por otro lado, para superar el segundo problema, tenemos que evitar incluir operaciones de aproximación en la definición de la nueva función de evaluación $g_{2t}^{1'}$ como hemos hecho con g_{2t}^1. Por tanto, podemos decir que los dos primeros problemas quedan resueltos con la definición de g_{2t}^1, con esto solo nos queda superar el tercer problema, para lo cual, tenemos que suavizar los grados de relevancia que obtienen g_{2t}^1 cuando el valor de umbral se satisface minimamente por el peso del término índice.

Como se mencionó anteriormente, la semántica de umbral simétrico tiene un comportamiento simétrico a ambos lados del valor de umbral medio, ya que está definida para distinguir dos situaciones en la interpretación del umbral:

- cuando el valor de umbral está a la izquierda del término central (pesos para indicar presencia del término en el documento), y
- cuando está a la derecha de este valor central (para indicar la ausencia).

Por consiguiente, analizando únicamente el caso de los pesos de presencia, podemos derivar rápidamente el comportamiento para el caso opuesto. Además, por simplicidad partiremos de la definición de $g^{1'}$.

Cuando el peso lingüístico de umbral s_b dado por un usuario es mayor, en el sentido habitual, que la etiqueta central $s_{\frac{T}{2}}$ del conjunto de términos $\mathcal{S}$, la función de evaluación $g^{1'}$ es no-decreciente. Como se dijo antes, $g^{1'}$ recompensaba en exceso a aquellos documentos cuyo valor $\mathcal{F}$ supera al peso de umbral s_b, y penaliza también en exceso a aquellos documentos que no sobrepasan s_b.

En esta Sección presentamos una función de evaluación $g_{2t}^{1'}$ que suaviza el comportamiento de $g^{1'}$. Concretamente, para conseguir este objetivo, $g_{2t}^{1'}$ debería trabajar como sigue: cuanto más sobrepase $\mathcal{F}$ el valor umbral y más cerca estén ambos del máximo valor de relevancia s_T, mayor serán los RSVs de los documentos. Por el contrario, cuando $\mathcal{F}$ sea menor que el peso de umbral y muy proximo al valor mínimo de relevancia s_0, más bajos será el RSV obtenido para el documento. En la literatura, a estas dos situaciones se las conoce como *sobresatisfacción* (oversatisfaction) y *infrasatisfacción* (undersatisfaction) [61].

Si suponemos un dominio numérico continuo [0,T], el la Figura 4.3, podemos ver gráficamente el comportamiento deseado para $g_{2t}^{1'}$ en tres situaciones diferentes, para tres valores de umbral dados $\frac{T}{2}$, u y u', siendo 0, $\frac{T}{2}$ y T los índices de los siguiente términos de $\mathcal{S}$: *término más bajo*, *término central* y *término mayor*, respectivamente.

Si nos centramos en el caso del valor umbral u (ver Figura 4.4), entonces dados dos

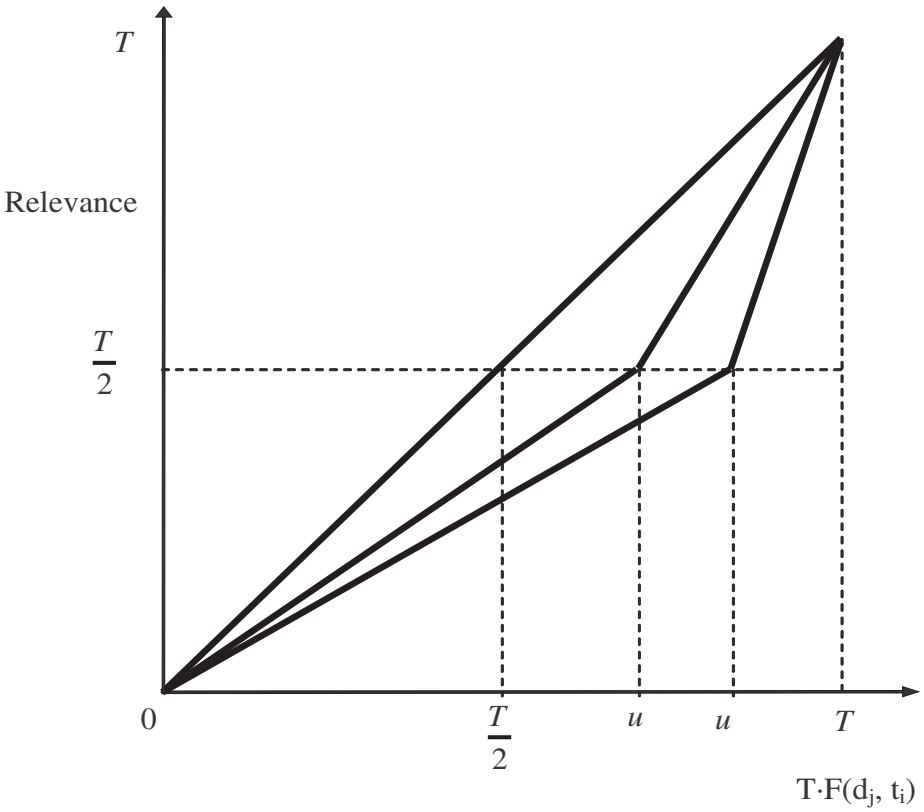

Figura 4.3: Comportamiento deseado de la función de evaluación $g_{2t}^{1'}$.

posibles valores de pesos de términos índice $a_1 < u$ y $a_2 > u$, los grados de relevancia obtenidos por la deseada función de evaluación deberían ser β_1 y $\beta_2 + \frac{T}{2}$. Asumiendo esta hipótesis, la definición de la función de evaluación $g_{2t}^{1'} : \mathcal{D} \times \mathcal{T} \times \mathcal{S} \longrightarrow (\mathcal{S} \times [-.5, .5))$ en el lado derecho del término central sería como sigue:

$$RSV_j^{i,1} = g_{2t}^{1'}(d_j, t_i, c_i^1) = \begin{cases} \Delta(\beta_2 + \frac{T}{2}) & ((s_a, \alpha_a) \geq (s_b, 0)) \wedge ((s_b, 0) \geq (s_{\frac{T}{2}}, 0)) \\ \Delta(\beta_1) & ((s_a, \alpha_a) < (s_b, 0)) \wedge ((s_b, 0) \geq (s_{\frac{T}{2}}, 0)) \end{cases}$$

donde $(s_a, \alpha_a) = \Delta(T \cdot \mathcal{F}(d_j, t_i))$ y $(s_b, 0)$ es la representación en el modelo lingüístico 2-tupla del peso lingüístico de umbral, c_i^1, dado por un usuario, y β_1 y β_2 son valores numéricos obtenidos como sigue. En la Figura 4.4, dos triángulos muestran el comportamiento de la función de evaluación deseada. El triángulo a la derecha del valor central $\frac{T}{2}$, muestra la manera en que son recompensados los documentos que tienen un peso del término índice a_2 mayor que un valor de umbral u, y el triángulo de la izquierda como son penalizados los documentos que tienen un peso del término índice a_1 menor que u. Analizando ambos triángulos podemos calcular las siguientes expresiones para β_2 y $\beta 1$:

$$\frac{T - \frac{T}{2}}{T - u} = \frac{\beta_2}{a_2 - u} \Longrightarrow \beta_2 = \frac{T \cdot (a_2 - u)}{2 \cdot (T - u)}$$

$$\frac{\frac{T}{2}}{u} = \frac{\beta_1}{a_1} \Longrightarrow \beta_1 = \frac{a_1 \cdot \frac{T}{2}}{u} = \frac{a_1 \cdot T}{2 \cdot u}$$

donde:

- $u = \Delta^{-1}(s_b, 0)$ siendo s_b el valor lingüístico de umbral proporcionado por un usuario,

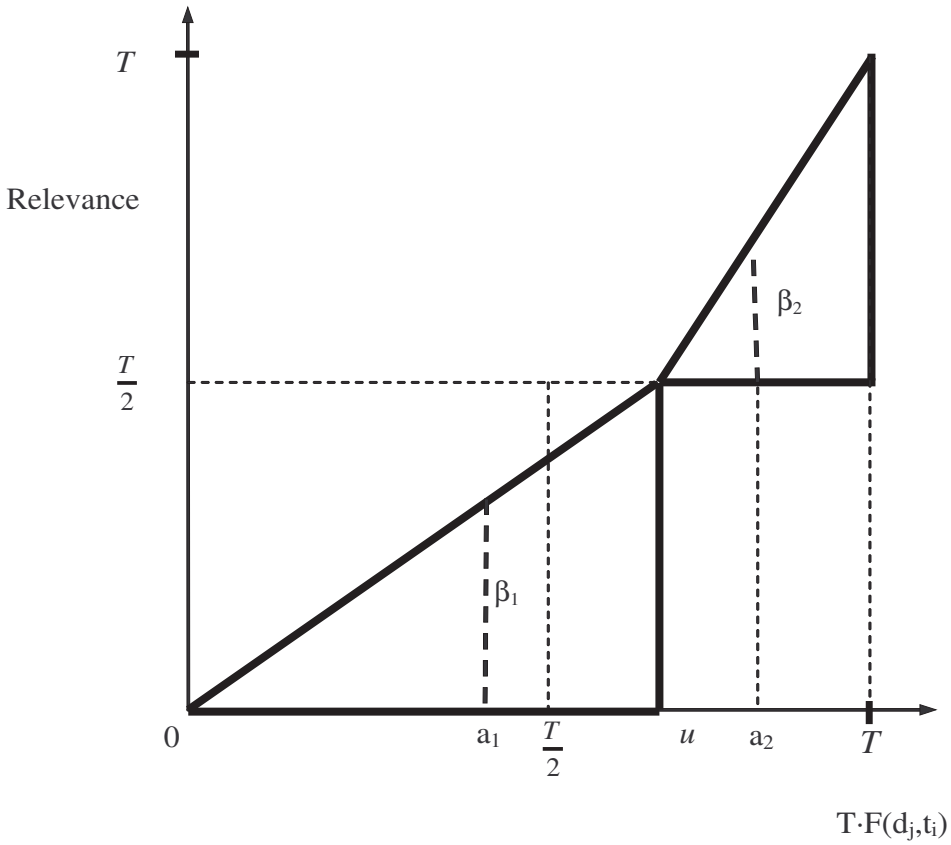

Figura 4.4: Comportamiento deseado de $g_{2t}^{1'}$ para valores umbral a la derecha del término central.

- a_2 debería ser el peso numérico de algún término índice t_i representado el contenido de un documento d_j, es decir, $a_2 = T \cdot \mathcal{F}(d_j, t_i)$, y similarmente,

- a_1 debería ser el peso numérico de algún término índice t_i representado el contenido de un documento d_k, es decir, $a_1 = T \cdot \mathcal{F}(d_k, t_i)$.

Resumiendo, dado que $g_{2t}^{1'}$, al igual que $g^{1'}$, debe presentar un comportamiento simétrico a ambos lados del valor de umbral medio, la definición completa de $g_{2t}^{1'}$ se obtiene fácilmente como sigue:

$$RSV_j^{i,1} = g_{2t}^{1'}(d_j, t_i, c_i^1) = \begin{cases} \Delta(\beta_2 + \frac{T}{2}) & ((s_a, \alpha_a) \geq (s_b, 0)) \wedge ((s_b, 0) \geq (s_{\frac{T}{2}}, 0)) \\ \Delta(\beta_1) & ((s_a, \alpha_a) < (s_b, 0)) \wedge ((s_b, 0) \geq (s_{\frac{T}{2}}, 0)) \\ \Delta(\beta_2^* + \frac{T}{2}) & ((s_a, \alpha_a) \leq (s_b, 0)) \wedge ((s_b, 0) < (s_{\frac{T}{2}}, 0)) \\ \Delta(\beta_1^*) & ((s_a, \alpha_a) > (s_b, 0)) \wedge ((s_b, 0) < (s_{\frac{T}{2}}, 0)) \end{cases}$$

donde $\beta_2 = \frac{T \cdot (a_2 - u)}{2 \cdot (T - u)}$, $\beta_1 = \frac{a_1 \cdot T}{2 \cdot u}$, $\beta_2^* = \frac{T \cdot (u - a_1)}{2 \cdot u}$, $\beta_1^* = \frac{T \cdot (T - a_2)}{2 \cdot (T - u)}$, $u = \Delta^{-1}(s_b, 0)$, $a_1 = T \cdot \mathcal{F}(d_k, t_i)$ y $a_2 = T \cdot \mathcal{F}(d_j, t_i)$ y $(s_b, 0)$ es la representación en el modelo lingüístico 2-tupla del peso de umbral, c_i^1, dado por el usuario.

Considerando el conjunto de etiquetas $\mathcal{S}$=$\{s_0 = N, s_1 = EL, s_2 = VL, s_3 = L, s_4 = M, s_5 = H, s_6 = VH, s_7 = EH, s_8 = T\}$, en las Tablas 4.5 y 4.6 mostramos una comparativa del comportamiento de ambas funciones de evaluación de la semántica de umbral simétrico, $g^{1'}$ y $g_{2t}^{1'}$ (ver columnas cuatro y seis) cuando $s_b \geq s_{\frac{T}{2}}$, esto es, para $s_b \in \{s_4, s_5, s_6, s_7, s_8\}$. Para realizar mejor la comparación, también mostramos los valores de $g_{2t}^{1'}$ proyectados en el domino lingüístico ordinal (ver columna cinco), esto es, considerando los resultados de $g_{2t}^{1'}$ en el domino lingüístico 2-tupla ($\mathcal{S} \times 0$). Por

tanto, analizando la definición de $g_{2t}^{1'}$ y los resultados mostrados en las Tablas 4.5 y 4.6, podemos destacar lo siguiente:

1. $g_{2t}^{1'}$ es no-decreciente para valores de umbral mayores que el valor de umbral medio y decreciente para valores de umbral menores que el valor de umbral medio, y por consiguiente, trabaja como la función de evaluación de umbral simétrico $g^{1'}$, siendo consistente con la semántica de umbral simétrico.

2. El problema de pérdida de precisión en los resultados se ha solucionado gracias al uso del modelo de representación lingüístico difuso 2-tupla, con el cual, $g_{2t}^{1'}$ produce resultados más precisos que $g^{1'}$, ya que al valor lingüístico obtenido se le asocia el valor numérico que mide la diferencia de información derivada, lo que se conoce como traslación simbólica [46]. Además, hay que apuntar que esta mejora en la precisión de los resultados puede ayudar a ordenar los documentos en la salida del SRI. Por ejemplo, en las filas 38 y 39 de la Tabla 4.6 $g^{1'}$ devuelve los mismos grados de relevancia, es decir, s_0 y s_0, mientras que $g_{2t}^{1'}$ obtiene $(s_1, -.5)$ y $(s_1, 0)$ respectivamente. Por tanto, en este caso, $g_{2t}^{1'}$ produce resultados más precisos y por consiguiente, permite ordenar mejor los documentos evaluados en las filas 38 y 39.

3. El problema de la pérdida de información en los resultados proporcionados por $g^{1'}$ se solventa también ya que no usamos operaciones de aproximación en su definición y la representación lingüística difusa 2-tupla permite recoger toda la información generada en el proceso de computación con palabras generados por la aplicación de $g_{2t}^{1'}$. Por ejemplo, en las Tablas 4.5 y 4.6 podemos observar que en muchos casos (filas, 11 a 14, 20 a 24, 29 a 34, 38, 40, 42, 44) si trabajamos con la función $g_{2t}^{1'}$ en un contexto lingüístico ordinal existe pérdida de información ya que no se representa el valor de la traslación simbólica.

4. Con respecto al problema de sobrevaloración de $g^{1'}$, podemos decir que $g_{2t}^{1'}$ consigue suavizar este comportamiento. Por ejemplo, si comparamos las expresiones de ambas funciones en el caso de valores de umbral por encima (a la derecha) del valor central y suponiendo que estamos en un caso en el que se satisface el umbral, los resultados obtenidos por $g_{2t}^{1'}$ están en el intervalo lingüístico 2-tupla $[(s_{\frac{T}{2}}, 0), (s_T, 0)]$ (usando la proyección de $g_{2t}^{1'}$ en un domino lingüístico ordinal $\mathcal{S}(g_{2t}^{1'}(\mathcal{S}))$), lo que significa que se obtienen del conjunto de etiquetas:

$$\{s_{\frac{T}{2}}, s_{\frac{T}{2}+1}, \ldots, s_T\},$$

mientras que los resultados obtenidos por $g^{1'}$ están determinados por el conjunto de etiquetas:

$$\{s_p = Label(\mathcal{F}(d_j, t_i)), s_{p+1}, \ldots, s_T\},$$

siendo $s_p = Label(\mathcal{F}(d_j, t_i))$ el peso lingüístico ordinal del término t_i que representa el contenido del documento d_j igual al valor de umbral deseado s_b y manteniendo la siguiente relación:

$$g^{1'}(d_j, t_i, s_b) \geq g_{2t}^{1'}(\mathcal{S})(d_j, t_i, s_b),$$

para todo $Label(\mathcal{F}(d_j, t_i)) \geq s_p \geq s_{\frac{T}{2}}$.

Este hecho puede observarse fácilmente en las Tablas 4.5 y 4.6. Igualmente, ocurre en el caso de no-satisfacción.

$\mathcal{D}$	$\mathcal{F}(d_j, t_i)$	s_b	$g^{1'}$	$g^{1'}_{2t}(\mathcal{S})$	$g^{1'}_{2t}$
1	s_0	s_4	s_0	s_0	$(s_0, 0)$
2	s_1	s_4	s_0	s_1	$(s_1, 0)$
3	s_2	s_4	s_1	s_2	$(s_2, 0)$
4	s_3	s_4	s_3	s_3	$(s_3, 0)$
5	s_4	s_4	s_4	s_4	$(s_4, 0)$
6	s_5	s_4	s_5	s_5	$(s_5, 0)$
7	s_6	s_4	s_7	s_6	$(s_6, 0)$
8	s_7	s_4	s_8	s_7	$(s_7, 0)$
9	s_8	s_4	s_8	s_8	$(s_8, 0)$
10	s_0	s_5	s_0	s_0	$(s_0, 0)$
11	s_1	s_5	s_0	s_1	$(s_1, -0.2)$
12	s_2	s_5	s_1	s_2	$(s_2, -0.4)$
13	s_3	s_5	s_2	s_2	$(s_2, 0.4)$
14	s_4	s_5	s_4	s_3	$(s_3, 0.2)$
15	s_5	s_5	s_5	s_4	$(s_4, 0)$
16	s_6	s_5	s_6	s_5	$(s_5, 0.33)$
17	s_7	s_5	s_8	s_7	$(s_7, -0.33)$
18	s_8	s_5	s_8	s_8	$(s_8, 0)$
19	s_0	s_6	s_0	s_0	$(s_0, 0)$
20	s_1	s_6	s_0	s_1	$(s_1, -0.33)$
21	s_2	s_6	s_1	s_1	$(s_1, 0.33)$
22	s_3	s_6	s_2	s_2	$(s_2, 0)$
23	s_4	s_6	s_3	s_3	$(s_3, -0.33)$
24	s_5	s_6	s_5	s_3	$(s_3, 0.33)$
25	s_6	s_6	s_6	s_4	$(s_4, 0)$
26	s_7	s_6	s_7	s_6	$(s_6, 0)$
27	s_8	s_6	s_8	s_8	$(s_8, 0)$
28	s_0	s_7	s_0	s_0	$(s_0, 0)$
29	s_1	s_7	s_0	s_1	$(s_1, -0.43)$
30	s_2	s_7	s_1	s_1	$(s_1, 0.14)$
31	s_3	s_7	s_2	s_2	$(s_2, 0.29)$
32	s_4	s_7	s_3	s_2	$(s_2, 0.29)$
33	s_5	s_7	s_4	s_3	$(s_3, 0.14)$
34	s_6	s_7	s_6	s_3	$(s_3, 0.43)$
35	s_7	s_7	s_7	s_4	$(s_4, 0)$
36	s_8	s_7	s_8	s_8	$(s_8, 0)$

Tabla 4.5: Comportamiento de las funciones de evaluación de la semántica de umbral simétrico.

$\mathcal{D}$	$\mathcal{F}(d_j, t_i)$	s_b	$g^{1'}$	$g_{2t}^{1'}(\mathcal{S})$	$g_{2t}^{1'}$
37	s_0	s_8	s_0	s_0	$(s_0, 0)$
38	s_1	s_8	s_0	s_1	$(s_1, -0.5)$
39	s_2	s_8	s_0	s_1	$(s_1, 0)$
40	s_3	s_8	s_2	s_2	$(s_2, -0.5)$
41	s_4	s_8	s_3	s_2	$(s_2, 0)$
42	s_5	s_8	s_4	s_3	$(s_3, -0.5)$
43	s_6	s_8	s_5	s_3	$(s_3, 0)$
44	s_7	s	s_7	s_4	$(s_4, -0.5)$
45	s_8	s_8	s_8	s_4	$(s_4, 0)$

Tabla 4.6: Comportamiento de las funciones de evaluación de la semántica de umbral simétrico (Continuación).

4.4.1. Ejemplo Teórico del Rendimiento del Nuevo Sistema de Recuperación de Información Ponderado Lingüístico 2-tupla con $g_{2t}^{1'}$

Para mostrar el rendimiento de $g_{2t}^{1'}$, desarrollaremos un ejemplo simplificado del modelo de SRI propuesto en la Sección 4.3, centrandonos solo en aquellas partes que quedarían afectadas por la nueva función $g_{2t}^{1'}$. Estas simplificaciones son:

- Suponemos que el usuario no introduce pesos con semántica cuantitativa, es decir, no está interesado en reducir el número de documentos recuperados para cada término t_i de la consulta. Por lo tanto, no es necesario aplicar el paso 3 (*Evaluación de los átomos con respecto a la semántica cuantitativa*) del SRI lingüístico 2-tupla propuesto en la Sección 4.3.

- También supondremos, que el usuario tampoco considera que unos términos de la consulta sea más importantes que otros, por lo que no introduce pesos asociado a la semántica de importancia relativa. De ahí que no sea necesario aplicar el paso 4 (*Evaluación de subexpresiones y modelado de la semántica de importancia*

relativa) del modelo de SRI propuesto en la Sección 4.3. Por lo tanto, no sería necesario elegir una función de transformación h ni utilizar un operador de agregación información ponderada, solo necesitaríamos usar el operador de agregación ϕ_{2t} oportuno.

Hechas estas puntualizaciones, empezamos el ejemplo. Supongamos una pequeña base de datos con siete documentos $\mathcal{D} = \{d_1, \ldots, d_7\}$, representados por un conjunto de diez términos índice $\mathcal{T} = \{t_1, \ldots, t_{10}\}$. Los documentos ha sido indizados por la función $\mathcal{F}$ obteniendo las siguientes representaciones:

$d_1 = 0.7/t_5 + 0.4/t_6 + 1/t_7$

$d_2 = 1/t_4 + 0.6/t_5 + 0.8/t_6 + 0.9/t_7$

$d_3 = 0.5/t_2 + 1/t_3 + 0.8/t_4$

$d_4 = 0.9/t_4 + 0.5/t_6 + 1/t_7$

$d_5 = 0.7/t_3 + 1/t_4 + 0.4/t_5 + 0.8/t_9 + 0.6/t_{10}$

$d_6 = 0.8/t_5 + 0.99/t_6 + 0.8/t_7$

$d_7 = 0.8/t_5 + 0.02/t_6 + 0.8/t_7 + 0.9/t_8$

Usando el conjunto de nueve etiquetas anterior, y considerando que un usuario formula la siguiente consulta:

$$q = ((t_5, VH) \vee (t_7, H)) \wedge ((t_6, L) \vee (t_7, H)).$$

El proceso de evaluación se desarrolla como sigue:

Evaluación de los átomos con respecto a la semántica de umbral simétrico

En este paso, primero representamos los documentos en el domino 2-tupla aplicando la función $\Delta(T \cdot \mathcal{F}(d_j, t_i))$.

$d_1 = (VH, -.4)/t_5 + (L, .2)/t_6 + (T, 0)/t_7$

$d_2 = (T, 0)/t_4 + (H, -.2)/t_5 + (VH, .4)/t_6 + (EH, .2)/t_7$

$d_3 = (M, 0)/t_2 + (T, 0)/t_3 + (VH, .4)/t_4$

$d_4 = (EH, .2)/t_4 + (M, 0)/t_6 + (T, 0)/t_7$

$d_5 = (VH, -.4)/t_3 + (T, 0)/t_4 + (L, .2)/t_5 + (VH, .4)/t_9 + (H, -.2)/t_{10}$

$d_6 = (VH, .4)/t_5 + (T, -.08)/t_6 + (VH, .4)/t_7$

$d_7 = (VH, .4)/t_5 + (N, .16)/t_6 + (VH, .4)/t_7 + (EH, .2)/t_8$.

Después evaluamos los átomos de acuerdo a la semántica de umbral simétrico por medio de $g_{2t}^{1'}$:

- para (t_5,VH):

$$\{RSV_1^5 = (\text{M,-.27}),\ RSV_2^5 = (\text{L,.2}),\ RSV_5^5 = (\text{VL,.13}),$$
$$RSV_6^5 = (\text{H,-.2}),\ RSV_7^5 = (\text{H,-.2})\}$$

- para (t_6,L):

$$\{RSV_1^6 = (\text{M,-.16}),\ RSV_2^6 = (\text{EL,.28}),\ RSV_4^6 = (\text{L,.2}),$$
$$RSV_6^6 = (\text{N,.06}),\ RSV_7^6 = (\text{T,-.16})\}$$

- y para (t_7,H):

$$\{RSV_1^7 = (\text{T,0}),\ RSV_2^7 = (\text{EH,-.07}),\ RSV_4^7 = (\text{T,0}),$$
$$RSV_6^7 = (\text{VH,-.13}),\ RSV_7^7 = (\text{VH,-.13})\}$$

siendo $RSV_j^i = g_{2t}^{1'}(d_j, t_i, (c_i, 0))$, y donde por ejemplo, el valor RSV_2^7 se calcula como:

$$RSV_2^7 = g_{2t}^{1'}(d_j, t_i, (c_i, 0)) = \Delta(\tfrac{8\cdot(7.2-5)}{2\cdot(8-5)} + \tfrac{8}{2}) = \Delta(6.93) = (s_7 = EH, -.07).$$

Evaluación de las subexpresiones. La consulta q tiene dos subexpresiones, $q_1 = (t_5, VH) \vee (t_7, H)$ y $q_2 = (t_6, L) \vee (t_7, H)$. Cada subexpresión está en forma disyuntiva, y por tanto, tenemos que aplicar un operador $LOWA_{2t}$ con $orness(W) > 0.5$ (por ejemplo, $W = [0.7, 0.3]$). En el resultado que obtenemos es el que sigue:

- para $q_1 = (t_5, VH) \vee (t_7, H)$:

$$\{RSV_1^1 = (\text{EH},\text{-}.28),\ RSV_2^1 = (\text{VH},\text{-}.19),\ RSV_4^1 = (\text{VH},\text{-}.4),$$
$$RSV_5^1 = (\text{EL},.49),\ RSV_6^1 = (\text{VH},\text{-}.45),\ RSV_7^1 = (\text{VH},\text{-}.45)\}$$

- y para $q_2 = (t_6, L) \vee (t_7, H)$:

$$\{RSV_1^2 = (\text{EH},\text{-}.25),\ RSV_2^2 = (\text{H},.24),\ RSV_4^2 = (\text{EH},\text{-}.44),$$
$$RSV_6^2 = (\text{M},.13),\ RSV_7^2 = (\text{EH},.25)\}$$

siendo ahora RSV_j^i el resultado de evaluar la subexpresión q_i con respecto al documento d_j, donde, por ejemplo, RSV_2^2 se obtiene como:

$$RSV_2^2 = \phi_{2t}^2(RSV_2^6 = (\text{EL},.28),\ RSV_2^7 = (\text{EH},\text{-}.07)) =$$
$$= \Delta(6.93 \cdot 0.7 + 1.28 \cdot 0.3) = \Delta(5, 24) = (H, .24),$$

de tal forma que $\Delta^{-1}(EL, .28) = 1.28$ y $\Delta^{-1}(EH, -.7) = (6.93)$.

Evaluación de la consulta completa. La consulta completa se evalúa usando el operador $LOWA_{2t}$ con un valor $orness(W) < 0.5$ (por ejemplo $W = [0.3, 0.7]$), ya que está expresada en forma conjuntiva. Haciendo esto, obtenemos el siguiente conjunto difuso de documentos:

$$\{RSV_1 = (\text{EH},\text{-}.27),\ RSV_2 = (\text{H},.41),\ RSV_4 = (\text{VH},\text{-}.11),$$
$$RSV_5 = (\text{N},.45),\ RSV_6 = (\text{H},\text{-}.44),\ RSV_7 = (\text{VH},.06)\}.$$

Para evaluar la mejora de rendimiento de un SRI al utilizar $g_{2t}^{1'}$, comparamos sus resultados con los que obtenemos por el mismo SRI en un marco lingüístico ordinal y utilizando la función de evaluación $g^{1'}$:

$$\{RSV_1 = \text{EH}, RSV_2 = \text{VH}, RSV_4 = \text{VH}, RSV_5 = \text{EL}, RSV_6 = \text{H}, RSV_7 = \text{H}\}.$$

Analizando estos resultados podemos concluir lo siguiente:

1. Primeramente, es obvio el beneficio de usar el modelo de representación lingüístico difuso 2-tupla, dado que si usamos una representación lingüística ordinal es imposible determinar la diferencia de relevancia entre algunos documentos, por ejemplo entre d_2 y d_4 o entre d_6 o d_7. Estos hechos son fácilmente observables usando el formato lingüístico 2-tupla.

2. Por otro lado, debemos apuntar que el SRI basado en la función de evaluación $g_{2t}^{1'}$ obtiene resultados más consistentes, que reflejan mejor los grados de relevancia de algunos términos con respecto a las necesidades de información expresadas por el usuario. Por ejemplo:

   - Si observamos la representación del documento d_5, vemos que éste no satisface ningún criterio expresado en la consulta ponderada q, es decir, no contiene los términos t_6 y t_7, y aunque contiene el término t_5, su peso es menor que el valor de umbral asociado con t_5 en la consulta. Por tanto, parece más razonable y consistente juzgar esta situación con un valor bajo de relevancia N que con un valor EL.
   - Si observamos la representación de los documentos d_1 y d_7, podemos ver que ambos presentan valores de satisfacción, con respecto a la consulta, muy similares, sin embargo, el SRI basado en $g^{1'}$ devuelve grados de relevancia más diferentes que en el caso del SRI que usa $g_{2t}^{1'}$.

4.4.2. Ejemplo Práctico del Rendimiento del Nuevo Sistema de Recuperación de Información Ponderado Lingüístico 2-tupla con $g_{2t}^{1'}$

En este apartado mostramos un ejemplo práctico de rendimiento del nuevo modelo de SRI propuesto en la Sección 4.3 utilizando la nueva función de interpretación de la semántica de umbral simétrico $g_{2t}^{1'}$, a este modelo de SRI lo notaremos por $SRI_{2t}^{'}$ para diferenciarlo del propuesto en 4.3. Los resultados de $SRI_{2t}^{'}$ serán comparados con los obtenidos por el modelo SRI_{2t} y mostrados en las Tablas 4.7 y 4.8.

En esta ocasión sólo nos centraremos en mostrar de manera práctica las bondades de la nueva función de evaluación de la semántica de umbral simétrico $g_{2t}^{1'}$. Para una comparación más fiel, usaremos las mismas consultas que en la Sección 4.3.3.

Para más ejemplos véase el Anexo B, en el cual se podrán encontrar también los documentos en los que aparecen los términos usados en los siguientes ejemplos.

	RESULTADOS DE LA CONSULTA	
	Parametros utilizados	
Jerarquia = (3,3)	*Base de Datos* = "trec"(5000 Docs.)	*orness* = _
	Numero de documentos recuperados = 13	
Rank	**ID Doc**	**RSV**
1#	4220	(EL,0.00)
2#	3030	(EL,0.00)
3#	4133	(EL,-0.19)
4#	4782	(EL,-0.33)
5#	4157	(EL,-0.40)
6#	4459	(EL,-0.41)
7#	2621	(EL,-0.44)
8#	4097	(EL,-0.48)
9#	4984	(EL,-0.49)
10#	185	(EL,-0.50)
11#	1816	(N,0.45)
12#	2423	(N,0.44)
13#	1980	(N,0.41)

Tabla 4.7: Evaluación de $\langle clamp, H, VL, \_\rangle$ con SRI'_{2t}.

	RESULTADOS DE LA CONSULTA	
	Parametros utilizados	
Jerarquia = (3,3)	*Base de Datos* = "trec"(5000 Docs.)	*orness* = 0.80
	Numero de documentos recuperados = 2	
Rank	**ID Doc**	**RSV**
1#	185	(N,0.32)
2#	2423	(N,0.27)

Tabla 4.8: Evaluación de $\langle bay, H, VL, \_\rangle AND \langle clamp, T, EL, \_\rangle$ con SRI'_{2t}.

4.5. Algunos Comentarios

En este capítulo, hemos presentado un nuevo modelo de SRI lingüístico difuso basado en el enfoque lingüístico difuso 2-tupla. Tal enfoque nos permite evitar los problemas de pérdida de precisión e información que padecen en su actividad los modelos de SRI lingüísticos ordinales. Cosecuentemente, todo esto mejora su rendimiento. Este mejora se consigue gracias a que la evaluación de la relevancia de los documentos no solo viene expresada por medio de una etiqueta lingüística, sino además de un valor de traslación que almacena cierta información. Información que se pierde en el caso de los modelos de SRI lingüísticos ordinales.

Adicionalmente, hemos incorporado un nuevo operador de agregación, el operador lingüístico LOWA 2-tupla, que nos permite suavizar el comportamiento de los conectores Booleanos AND y OR.

Por último, y no por ello menos importante, hemos estudiado y refinado la interpretación de la semántica de umbral simétrico, relajando su comportamiento optimista/pesimista.

Todo esto contribuye a mejorar los resultados en el proceso de recuperación de información como lo atestiguan los ejemplos teóricos y prácticos desarrollados.

Capítulo 5

Un Nuevo Modelo de Sistema de Recuperación de Información con Información Lingüística no Balanceada

La mayoría de los SRI basados en enfoques lingüísticos usan conjuntos de etiquetas distribuidos de manera simétrica y uniforme para establecer los pesos de las consultas y los grados de relevancia de los documentos.

El problema es que al usar conjuntos de etiquetas distribuidos simétrica y uniformemente entorno a una etiqueta central, encontramos los mismos niveles de discriminación a ambos lados de ésta.

Sin embargo, los usuarios suelen buscar documentos con criterios positivos y por tanto, al formular sus consultas ponderadas introducen más pesos que están por encima de la etiqueta central que por debajo. Además, de una manera generalizada, los usuarios prestan más atención a los documentos relevantes recuperados que a los no relevantes.

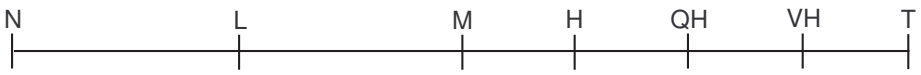

Figura 5.1: Ejemplo de un conjunto no balanceado de 7 etiquetas lingüísticas.

Por todo esto, parece lógico que, para mejorar la interacción entre el usuario y el sistema, éste fuera capaz de permitir escalas lingüísticas no balanceadas, es decir, conjunto de etiquetas, $\mathcal{S}_{un}$, con diferentes niveles de discriminación a ambos lados de la etiqueta central, como se muestra en la Figura 5.1, permitiendo que tanto el usuario, en la formulación de sus consultas ponderadas, como la salida final del sistema usen estos conjuntos de etiquetas.

En este capítulo, presentamos un modelo de SRI que acepta consultas ponderadas cuyos pesos se expresan mediante conjuntos de etiquetas no balanceadas. Este sistema permite además clasificar los documentos recuperados en clases de relevancia lingüística sobre conjuntos de etiquetas no balanceados.

Este capítulo se estructura como sigue: primero introduciremos todas las herramientas de modelado lingüístico necesarias para manejar información lingüística no balanceada y después pasaremos a describir el sistema propuesto en detalle, por último mostraremos ejemplos de funcionamiento, tanto en el marco teórico como en entornos prácticos.

5.1. Preliminares

En esta sección presentamos los conceptos que necesitamos para diseñar una metodología para manejar información lingüística no balanceada.

5.1.1. Jerarquías Lingüísticas Basadas en el Modelo 2-tupla

Las jerarquías lingüísticas fueron introducidas en [26] para mejorar el modelado lingüístico en sistemas difusos. También se utilizaron en [49] para mejora la precisión de los procesos de computación con palabras en contextos difusos multigranulares. En este capítulo usaremos esta herramienta para manejar conjuntos de términos lingüísticos no balanceados.

Como se vio en la Sección 3.6, una jerarquía lingüística es un conjunto de niveles, donde cada nivel representa un conjunto de términos lingüísticos con diferente granularidad unos de otros. Cada nivel se denota como $l(t, n(t))$, donde,

1. t es un número que indica el nivel de la jerarquía, y

2. $n(t)$ denota la granularidad del conjunto de términos lingüísticos del nivel t.

Asumimos niveles con términos lingüísticos cuya función de pertenencia es de forma triangular, y están simétrica y uniformemente distribuidas en $[0, 1]$. Además estos niveles tienen cardinalidad impar.

Los niveles dentro de la jerarquía se ordenan de acuerdo a su granularidad, es decir, para dos niveles consecutivos t y $t + 1$, $n(t + 1) > n(t)$. Por tanto, podemos entender que el nivel $t + 1$ es un refinamiento del nivel previo t.

Juntando todos estos conceptos, definimos una jerarquía lingüística, LH, como la unión de todos los niveles t:

$$LH = \bigcup_t l(t, n(t)).$$

Dada una jerarquía LH, denotamos como $\mathcal{S}^{n(t)}$ el conjunto de términos lingüísticos de

LH correspondientes al nivel t de LH caracterizados por una granularidad $n(t)$:

$$\mathcal{S}^{n(t)} = \{s_0^{n(t)}, \ldots, s_{n(t)-1}^{n(t)}\}.$$

Generalmente, podemos decir que el conjunto de términos lingüísticos del nivel $t+1$ se obtiene del predecesor como:

$$l(t, n(t)) \rightarrow l(t+1, 2 \cdot n(t) - 1).$$

Podemos ver un ejemplo gráfico de una jerarquía lingüística en la Figura 3.9. En esta figura tenemos tres niveles, uno con tres, otro con 5 y un último con 9 etiquetas respectivamente. Estos conjuntos son:

- l(1,3) = {N, M, T},
- l(2,5) = {N, L, M, H, T},
- l(3,9) = {N, VL, QL, L, M, H, QH, VH, T}.

En [49] se desarrollaron funciones de transformación entre etiquetas de diferentes niveles para realizar procesos de computación con palabras sin pérdida de información.

Definición 5.1. Sea $LH = \bigcup_t l(t, n(t))$ una jerarquía lingüística cuyos conjuntos de términos se denotan como $\mathcal{S}^{n(t)} = \{s_0^{n(t)}, \ldots, s_{n(t)-1}^{n(t)}\}$, y consideremos el modelo de representación de información lingüística 2-tupla introducido en la Sección 3.5. La transformación de una etiqueta del nivel t a una etiqueta del nivel $t^{'}$ se define como :

$$\tau_{t'}^{t} : l(t, n(t)) \longrightarrow l(t', n(t'))$$

$$\tau_{t'}^{t}(s_i^{n(t)}, \alpha^{n(t)}) =$$
$$\Delta_{n(t')}\left(\frac{\Delta_{n(t)}^{-1}(s_i^{n(t)}, \alpha^{n(t)}) \cdot (n(t') - 1)}{n(t) - 1}\right).$$

Proposition 5.1. *La función de transformación entre términos lingüísticos en diferentes niveles de la jerarquía es biyectiva:*

$$\tau_t^{t'}(\tau_{t'}^t(s_i^{n(t)}, \alpha^{n(t)})) = (s_i^{n(t)}, \alpha^{n(t)}).$$

5.1.2. Metodología para Manejar Información Lingüística no Balanceada

En esta subsección introducimos una metodología para manejar conjuntos de términos lingüísticos no balanceados usando el modelo de representación de información lingüística 2-tupla. Básicamente, esta metodología consiste en representar términos lingüísticos de diferentes niveles de LH, y realizar las operaciones de cómputo sobre estas etiquetas usando el modelo computacional del modelado lingüístico difuso 2-tupla introducido en la Sección 3.5.2.

Esta metodología presenta los siguientes pasos:

Representación de un Conjunto de Términos no Balanceado, $\mathcal{S}_{un}$, por medio de una Jerarquía Lingüística LH

Para hacer esto, usamos diferentes niveles de la jerarquía lingüística LH para representar ambos lados del término lingüístico central. Así, el lado con más términos lingüísticos necesitará un nivel con mayor granularidad $l(i, n(i))$ de LH y el lado con menos términos usará un nivel menos granular $l(j, n(j))$ de LH, siendo $i > j$. Concretamente, los pasos son:

1. elección de un nivel t^- con una adecuada granularidad para representar, usando el model 2-tupla, el subconjunto de términos lingüísticos del lado izquierdo de la etiqueta central de $\mathcal{S}_{un}$, y

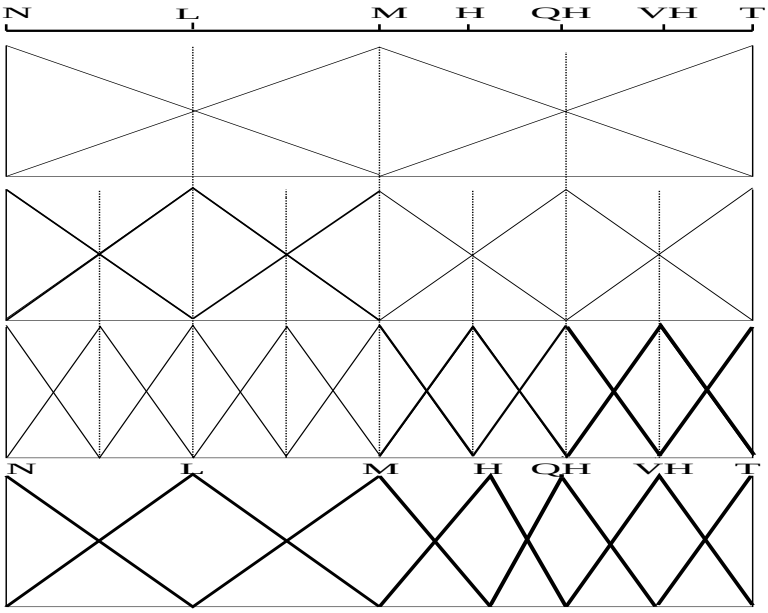

Figura 5.2: Jerarquía lingüística para representar un conjunto no balanceado de 7 etiquetas.

2. elección de un nivel t^+ con una adecuada granularidad para representar, usando el model 2-tupla, el subconjunto de términos lingüísticos de $\mathcal{S}_{un}$ del lado derecho de la etiqueta central.

Asumiendo el conjunto no balanceado de términos, $\mathcal{S}_{un} = \{N, L, M, H, QH, VH, T\}$, mostrado en la Figura 5.1 y la jerarquía lingüística de la Figura 3.9, mostramos un ejemplo de jerarquía lingüística en la que usamos diferentes niveles para representar los términos de ambos lados del término central (Figura 5.2) de $\mathcal{S}_{un}$. Así, para representar los términos $\{N, L, M\}$ usamos el nivel $l(2, n(2))$, $(t^- = l(2, n(2)))$, y para representar $\{H, QH, VH, T\}$ el nivel más adecuado es $l(3, n(3))$, $(t^+ = l(3, n(3)))$.

Modelo Computacional para Manejar Información Lingüística no Balanceada

Para manejar información lingüística no balanceada necesitamos un conjunto de herramientas de cálculo, así, en los siguientes puntos describiremos algunas herramientas básicas:

1. Elegimos un nivel $t' \in \{t^-, t^+\}$, tal que $n(t') = max\{n(t^-), n(t^+)\}$.

2. Comparación de dos 2-tupla no balanceadas $(s_k^{n(t)}, \alpha_1)$, $t \in \{t^-, t^+\}$, y $(s_l^{n(t)}, \alpha_2)$, $t \in \{t^-, t^+\}$, cada una representando una cantidad de información no balanceada. Su expresión es similar a la usada para comparar dos 2-tupla (Sección 3.5.2) pero actuando sobre los valores $\tau_{t'}^t(s_k^{n(t)}, \alpha_1)$ y $\tau_{t'}^t(s_l^{n(t)}, \alpha_2)$. Deberíamos apuntar que usando la comparación de dos 2-tupla no balanceadas podemos fácilmente definir los operadores de comparación Max_{un} and Min_{un}.

3. También necesitamos un operador de negación de información lingüística no balanceada:

   Definición 5.2. Sea $(s_k^{n(t)}, \alpha)$, $t \in \{t^-, t^+\}$ una 2-tupla no balanceada, entonces:

   $$\mathcal{NEG}(s_k^{n(t)}, \alpha) = Neg(\tau_{t''}^t(s_k^{n(t)}, \alpha)),$$

   $t \neq t'', t'' \in \{t^-, t^+\}$.

4. Por último también necesitamos definir una serie de operadores de agregación de información lingüística no balanceada . Esto se hace usando los operadores de agregación diseñados para manejar información lingüística en el modelo 2-tupla (Sección 3.5.2) pero actuando sobre los valores lingüísticos no balanceados transformados por medio de $\tau_{t'}^t$ que se introdujo en la Definición 3.16 de la Sección 3.6.

Después, una vez que el resultado ha sido obtenido, es transformado al correspondiente nivel $t \in \{t^-, t^+\}$ de LH por medio de $\tau_t^{t'}$ para expresar el resultado en el conjunto no balanceado de términos $\mathcal{S}_{un}$.

Por ejemplo, fácilmente podemos extender el operador $LOWA_{2t}$ definido en Sección 4.3.1, para trabajar con información lingüística no balanceada, a este operador lo denotamos como $LOWA_{un}$ y se define como sigue:

Definición 5.3. Sea $\{(a_1, \alpha_1), \ldots, (a_m, \alpha_m)\}$ un conjunto no balanceado de información lingüística 2-tupla a agregar, entonces el operador $LOWA_{un}$ se define como:

$$\phi_{un}((a_1, \alpha_1), \ldots, (a_m, \alpha_m)) = W \cdot B^T =$$

$$= C_{un}^m\{w_w, b_w, k = 1, \ldots, m\} =$$

$$= w_1 \otimes b_1 \oplus (1 - w_1) \otimes C_{un}^{m-1}\{\beta_h, b_h, h = 2, \ldots, m\}$$

donde $b_i = (a_i, \alpha_i) \in (\mathcal{S}_{un} \times [-.5, .5)), W = [w_1, \ldots, w_m]$, es un vector de pesos, tal que, $w_i \in [0, 1]$ and $\sum_i w_i = 1$, $\beta_h = \frac{w_h}{\sum_2^m w_k}, h = 2, \ldots, m$, y B es el vector ordenado de 2-tupla no balanceado. Cada elemento $b_i \in B$ es el i-esimo mayor de la colección $\{(a_1, \alpha_1), \ldots, (a_m, \alpha_m)\}$, y C_{un}^m es el operador de combinación convexa de m 2-tupla no balanceadas. Si $w_j = 1$ y $w_i = 0$ con $i \neq j \forall i, j$ la combinación convexa se define como: $C_{un}^m\{w_i, b_i, i = 1, \ldots, m\} = b_j$. Y si $m = 2$ entonces se define como:

$$C_{un}^2\{w_l, b_l, l = 1, 2\} = w_1 \otimes b_j \oplus (1 - w_1) \otimes b_i = \tau_t^{t'}(s_k^{n(t')}, \alpha)$$

donde $(s_k^{n(t')}, \alpha) = \Delta(\lambda)$ y $\lambda = \Delta^{-1}(\tau_{t'}^t(b_i)) + w_1 \cdot (\Delta^{-1}(\tau_{t'}^t(b_j)) - \Delta^{-1}(\tau_{t'}^t(b_i)))$, $b_j, b_i \in (\mathcal{S}_{un} \times [-.5, .5))$, $(b_j \geq b_i)$, $\lambda \in [0, n(t') - 1], t \in \{t^-, t^+\}$.

También podemos definir un operador ponderado de información lingüística no balanceada.

Generalmente, como vimos en en la Sección 4.3.1 un operador de agregación ponderado lleva a cabo dos actividades [42]: i) la transformación de la información ponderada bajo los grados ponderados, y ii) la agregación de la información ponderada transformada por medio de un operador de agregación de información no ponderada.

a) La transformación de la información ponderada bajo los grados ponderados por medio de la función de transformación h. Algunos familias de conectivos usados como funciones de transformación en un entorno no balanceado son los siguientes dos:

1) *Funciones de conjunción lingüística ($LC^{\rightarrow}$).* Las funciones conjuntivas que usaremos son las siguientes t-normas, las cuales son monotonamente no decrecientes en los pesos y satisfacen las propiedades requeridas por cualquier función de transformación, h, [36]:

- Clásico operador MIN:

$$LC_1^{\rightarrow}(\omega, a) = MIN_{un}(\omega, a),$$

donde MIN_{un} es el operador de comparación de 2-tupla no balanceadas.

- Nilpotent MIN:

$$LC_2^{\rightarrow}(\omega, a) = \begin{cases} MIN_{un}(\omega, a) & \text{if } \omega > \mathcal{NEG}(a) \\ s_0 & \text{en otro caso.} \end{cases}$$

- Conjunción débil:

$$LC_3^{\rightarrow}(\omega, a) = \begin{cases} MIN_{un}(\omega, a) & \text{if } MAX_{un}(\omega, a) = s_T \\ s_0 & \text{en otro caso.} \end{cases}$$

donde $\omega, a \in \mathcal{S}_{un} \times [-.5, .5)$, a es parte de la información a agregar y ω es el peso asociado a a.

2) *Funciones de implicación lingüística ($LI^{\rightarrow}$).* Las funciones de implicación lingüística que usaremos son monotonamente no crecientes en los pesos y satisfacen las propiedades requeridas para cualquier función de transformación h [36]:

- Función de implicación de Kleene-Dienes:

$$LI_1^{\rightarrow}(\omega, a) = MAX_{un}(\mathcal{NEG}(\omega), a),$$

donde MAX_{un} es el operador de comparación de 2-tupla no balanceadas.

- La función de implicación de Gödel:

$$LI_2^{\rightarrow}(\omega, a) = \begin{cases} s_T & \text{if } \omega \leq a \\ a & \text{en otro caso.} \end{cases}$$

- Función de implicación de Fodor:

$$LI_3^{\rightarrow}(\omega, a) = \begin{cases} s_T & \text{if } \omega \leq a \\ MAX_{un}(\mathcal{NEG}(\omega), a) & \text{en otro caso.} \end{cases}$$

donde $\omega, a \in \mathcal{S}_{un} \times [-.5, .5)$, a es parte de la información lingüística a agregar y ω es el peso asociado a a.

b) La agregación de la información ponderada transformada por medio de un operador de agregación de información no ponderada f. Como es sabido, la

elección de h depende de la de f.

Como operador f podemos usar $LOWA_{un}$.

5.2. Un Nuevo Modelo de Sistema de Recuperación de Información con Información Lingüística no Balanceada

En esta Sección presentamos un SRI lingüístico que usa un conjunto no balanceado de términos $\mathcal{S}_{un}$ para expresar juicios lingüísticos en el proceso de recuperación. Particularmente, $\mathcal{S}_{un}$ presenta un mayor número de niveles de discriminación a la derecha del término central que a la izquierda del mismo (por ejemplo como ocurre en el ejemplo de la Figura 5.1). Este SRI acepta consultas ponderadas lingüísticamente y proporciona los grados de relevancia (RSVs) sobre el conjunto de términos no balanceados $\mathcal{S}_{un}$ y $\mathcal{S}_{un} \times [-.5, .5)$, respectivamente. Los componentes de este SRI son los que se detallan en las siguientes subsecciones.

5.2.1. Base de Datos Documental

Como en la Sección 4.2, la *base de datos* almacena el conjunto finito de documentos $\mathcal{D} = \{d_1, \ldots, d_m\}$ y el conjunto finito de términos índice $\mathcal{T} = \{t_1, \ldots, t_l\}$. Los documentos son representados por medio de los términos índices, los cuales describen el contenido subyacente de los documentos. Existe una función de indexación $\mathcal{F} : \mathrm{D} \times \mathrm{T} \rightarrow [0, 1]$, la cual pondera los términos índice de acuerdo a su significancia a la hora de representar el contenido de los documentos con objeto de mejorar la recuperación de éstos. $\mathcal{F}(d_j, t_i) = 0$ significa que el documento d_j no está representado en absoluto por el/los

concepto/s que representa el término t_i y $\mathcal{F}(d_j, t_i) = 1$ implica que el documento d_j está perfectamente representado por el/los concepto/s indicados por t_i. De esta forma, podemos ver cada documento d_j representado como $R_{d_j} = \Sigma_{i=1}^{l} \mathcal{F}(d_j, t_i)/t_i$.

Asumimos también que el sistema usa alguno de métodos de ponderación existentes [83] para $\mathcal{F}$.

5.2.2. El Subsistema de Consulta

El *subsistema de consulta* presenta un lenguaje de consulta Booleano ponderado con el que el usuario expresa sus necesidades de información. En las consultas, los términos pueden ser ponderados de acuerdo a dos posibilidades semánticas, incluso de manera simultánea. Estás semánticas son la *semántica de umbral* y la semántica de *importancia relativa*. Como en [13] usamos la variable lingüística *Importance* para expresar los pesos lingüísticos asociados a los términos de la consulta. Además, consideramos un conjunto no balanceado de valores lingüísticos $\mathcal{S}_{un}$.

Al asociar pesos de umbral a los términos de la consulta, el usuario pide ver todos aquellos documentos que traten suficientemente el tema representado por tales términos. Con los pesos lingüísticos de importancia, el usuario pide ver todos los documentos cuyo contenido este más asociado a los conceptos representados por el término más importante que al resto de términos. Cada consulta se construye como una combinación de términos índice ponderados, que a su vez son conectados por medio de los operadores lógicos AND ($\wedge$), OR ($\vee$), y NOT ($\neg$).

Por tanto, una consulta $\mathcal{Q}$ es cualquier expresión Booleana cuyos componentes

atómicos son 3-tuplas de la forma $\langle t_i, c_i^1, c_i^2 \rangle$ perteneciendo al conjunto, $\mathcal{T} \times \mathcal{S}_{un}^2$; $t_i \in \mathcal{T}$, y c_i^1 y c_i^2 son valores lingüísticos de la variable lingüística *Importance* modelando la semántica de umbral (indicando la importancia que el término t_i debe tener en los documentos deseados) y la semántica de importancia relativa (indicado la importancia que el significado de t_i debe tener en el conjunto de documentos recuperados), respectivamente. Según esto, el conjunto $\mathcal{Q}$ de consultas legítimas se define por medio de las siguientes reglas sintácticas:

1. $\forall q = \langle t_i, c_i^1, c_i^2 \rangle \in \mathcal{T} \times \mathcal{S}^2 \longrightarrow q \in \mathcal{Q}$.
2. $\forall q, p \in \mathcal{Q} \longrightarrow q \wedge p \in \mathcal{Q}$.
3. $\forall q, p \in \mathcal{Q} \longrightarrow q \vee p \in \mathcal{Q}$.
4. $\forall q \in \mathcal{Q} \longrightarrow \neg q \in \mathcal{Q}$.
5. Todas las consultas legítimas $q \in \mathcal{Q}$ son solo aquellas obtenidas aplicando las reglas 1-4, inclusive.

5.2.3. El Subsistema de Evaluación

El objetivo del *subsistema de evaluación* es evaluar documentos en términos de su relevancia con respecto a las consultas Booleanas ponderadas según las dos posibilidades semánticas comentadas. Una consulta Booleana con más de un término ponderado se evalúa por medio de un proceso constructivo ascendente que incluye los siguientes pasos:

1. *Preprocesamiento de la consulta*: en este paso, la consulta del usuario se preprocesa y se transforma en otra que bien puede estar expresada en *forma normal conjuntiva* (CNF) o en *forma normal disyuntiva* (DNF), dando como resultado que todos sus subexpresiones Booleanas tienen al menos dos átomos.

2. *Evaluación de los átomos con respecto a la semántica de umbral*: en este paso, se evalúan los documentos con respecto a su relevancia a átomos individuales en la consulta, considerando solo las restricciones impuestas por la semántica de umbral. Según está semántica, asociando pesos de umbral a los términos de una consulta, el usuario busca aquellos documentos cuyo contenido esté suficientemente representado por tales términos. Para modelar la interpretación de la esta semántica de umbral, usamos la función de evaluación descrita en [77] pero definida en un contexto lingüístico 2-tupla, a esta función la denotamos por g_{un}, y la definimos como:

$$g_{un} : \mathcal{D} \times \mathcal{T} \times \mathcal{S} \longrightarrow \mathcal{S} \times [-.5, .5).$$

Entonces, dado un átomo $\langle t_i, c_i^1, c_i^2 \rangle$, $t_i \in \mathcal{T}$, y $d_j \in \mathcal{D}$, con g_{un} obtenemos el grado de relevancia parcial, RSV, en contexto 2-tupla de d_j, llamado $RSV_j^{i,1}$, midiendo cómo de bien el término índice ponderado $\mathcal{F}(d_j, t_i)$ satisface las restricciones impuestas a través del peso de umbral c_i^1 de acuerdo a la siguiente expresión:

$$g_{un}(d_j, t_i, c_i^1) = \begin{cases} (s_a, \alpha_a) & \text{if } (s_a, \alpha_a) \geq (c_i^1, 0) \\ \Delta(0) & \text{en otro caso.} \end{cases}$$

donde $(s_a, \alpha_a) = \Delta((n(t) - 1) \cdot \mathcal{F}(d_j, t_i))$, $\Delta : [0, n(t) - 1] \longrightarrow \mathcal{S} \times [-.5, .5)$ con $t = t^-$ si $\mathcal{F}(d_j, t_i) \leq .5$ y $t = t^+$ si $\mathcal{F}(d_j, t_i)\rangle.5$, siendo t^- y t^+ los niveles de LH.

3. *Evaluación de subexpresiones y modelado de la semántica de importancia relativa*: consideramos que la semántica de importancia relativa cuando tenemos un único átomo no tiene sentido. Por tanto, en este paso evaluamos la relevancia de los documentos con respecto a todas las subexpresiones de las consultas preprocesadas, que estarán compuestas por un número mínimo de dos componentes atómicos.

Dada una subexpresión q_v, con $\eta \geq 2$ átomos, sabemos que cada documento d_j presenta un grado de relevancia partial $RSV_j^{i,1} \in (\mathcal{S}_{un} \times [-.5, .5))$ con respecto a cada átomo $\langle t_i, c_i^1, c_i^2 \rangle$ de q_v. Entonces, la evaluación del grado de relevancia de un documento d_j con respecto a la consulta completa q_v implica la agregación de los grados de relevancia parcial $\{RSV_j^{i,1}, i = 1 \ldots, \eta\}$ ponderados por medio los pesos de importancia relativa $\{c_i^2 \in \mathcal{S}_{un}, i = 1, \ldots, \eta\}$. Para hacer esto, necesitamos un operador de agregación de información ponderada sobre 2-tupla que garantice que los términos más importante de la consulta sean los términos más influyentes en la determinación de los RSV.

En [92], Yager estudió el efecto de los grados de importanciá sobre los operadores de agregación tipo MAX y MIN y sugirió una clase de funciones de transformación de información para ambos tipos de agragación. Para la agregación tipo MIN, sugirió un conjunto de t-conormas actuando sobre la información ponderada y la negación de los grados de importancia, para las agregaciones tipo MAX sugirió una familia de t-normas actuando sobre la información ponderada y los grados de importancia. Como es sabido, la evaluación de los conectivos lógicos AND y OR por medio de los operadores MIN y MAX presenta algunas limitaciones. Esto es, pueden tener un comportamiento muy restrictivo e inclusivo respectivamente. Este hecho provoca que el proceso de recuperación pueda ser engañoso ya que, por un lado, la t-norma lingüística MIN puede provocar el rechazo de documentos útiles por la no satisfacción de alguno de los criterios simples de la subexpresión conjuntiva, y por otro lado, la t-conorma MAX puede provocar la aceptación de documentos inútiles por la simple satisfacción de alguno de los criterios.

Por tanto, para agregar información ponderada no balanceada podemos usar el operador $LOWA_{un}$, ϕ_{un}, junto con las funciones de transformación $LC_1^{\rightarrow}$ y $LI_1^{\rightarrow}$, para modelar los conectivos Booleanos AND y OR ponderados respectivamente. Además, estos operadores superan las limitaciones de las t-normas y t-conormas lingüísticas MIN y MAX ya que su comportamiento puede ser suavizado por medio del vector de pesos.

Por tanto, usamos la medida orness para controlar el comportamiento del operador $LOWA_{un}$, ϕ_{un}. En particular, proponemos usar un operador no balanceado ϕ_{un}^1 con $orness(W) \geq .5$ para modelar los conectivos AND y un operador no balanceado ϕ_{un}^2 con $orness(W) < .5$ para modelar el conectivo OR.

Por lo tanto, para evaluar subexpresiones junto con la semántica de importancia relativa y según las actividades necesarias para agregar información ponderada, si la subexpresión es conjuntiva entonces usamos $h = \phi_{un}^1$ y $f = MAX_{un}(\mathcal{NEG}(weight, 0), valor\_no\_balanceado)$, y si es disyuntiva entonces usamos $h = \phi_{un}^2$, con $f = MIN_{un}((weight, 0)$, valor_no_balanceado).

De manera resumida, dado un documento d_j, evaluamos su relevancia con respecto a una subexpresión q_v, llamada $RSV_j^v \in (\mathcal{S}_{un} \times [-.5, .5))$ como:

a) Si q_v es una subexpresión conjuntiva entonces:

$$RSV_j^v = \phi_{un}^1(MAX_{un}(\mathcal{NEG}(c_1^2, 0), RSV_j^{1,1}),$$

$$\ldots, MAX_{un}(\mathcal{NEG}(c_\eta^2, 0), RSV_j^{\eta,1})).$$

b) Si q_v es una subexpresión disyuntiva entonces:

$$RSV_j^v = \phi_{un}^2(MIN_{un}((c_1^2, 0), RSV_j^{1,1}),$$

$$\ldots, MIN_{un}((c_\eta^2, 0), RSV_j^{\eta,1})).$$

4. *Evaluación de la consulta completa:* En este paso final de la evaluación, los documentos son evaluados con respecto a su relevancia a combinaciones Booleanas en todas las subexpresiones existentes en una consulta. Para hacer esto, usamos de nuevo los operadores ϕ_{un}^1 y ϕ_{un}^2 para modelar los conectivos AND y OR, respectivamente.

   Entonces, dado un documento d_j, su relevancia con respecto a una consulta q, $RSV_j \in (\mathcal{S}_{un} \times [-.5, .5))$ se calcula como:

   a) Si q está en CNF entonces $RSV_v = \phi_{un}^1(RSV_j^1, \ldots, RSV_j^v)$

   b) Si q está en DNF entonces $RSV_v = \phi_{un}^2(RSV_j^1, \ldots, RSV_j^v)$,

   siendo v el número de subexpresiones de q.

NOTA: Sobre el operador NOT. Deberíamos notar que, si una consulta está en CNF o DNF, tenemos que definir el operador de negación solo a nivel del átomo simple. Esto simplifica la definición del operador NOT. Como se hizo en [51], la evaluación de un documento d_j ante un átomo negado $\langle \neg t_i, c_i^1, c_i^2 \rangle$ se obtiene de la negación del término índice ponderado $\mathcal{F}(t_i, d_j)$. Esto significa computar la función de evaluación del umbral g_{un} a partir del valor lingüístico no balanceado $\mathcal{NEG}(\Delta((n(t)-1)\cdot\mathcal{F}(d_j, t_i)))$, con $t = t^-$ si $\mathcal{F}(d_j, t_i) \leq .5$ y $t = t^+$ si $\mathcal{F}(d_j, t_i) > .5$.

De manera esquemática, este subsistema de evaluación puede ser sintetizado por medio de una función de evaluación lingüística general $\mathcal{E}_{un} : \mathcal{D} \times \mathcal{Q} \longrightarrow \mathcal{S}_{un} \times [-.5, .5))$, la cual evalúa las diferentes clases de consultas preprocesadas, $\{q = \langle t_i, c_i^1, c_i^2 \rangle, q \wedge p, q \vee p\}$ de acuerdo a las siguientes cinco reglas:

1. *Átomos*:

$$\mathcal{E}_{un}(d_j, q^1) = g_{un}(d_j, t_i, c_i^1),$$

tal que, $q^1 = \langle t_i, c_i^1, c_i^2 \rangle$.

2. *Subexpresiones Conjuntivas*:

$$\mathcal{E}_{un}(d_j, q^2) = \phi_{un}^1(MAX_{un}(\mathcal{NEG}(c_1^2, 0), \mathcal{E}_{un}(d_j, q_1^1)),$$

$$\ldots, MAX_{un}(\mathcal{NEG}(c_\eta^2, 0), \mathcal{E}_{un}(d_j, q_\eta^1))),$$

siendo η el número de átomos de q^2.

3. *Subexpresiones Disyuntivas*:

$$\mathcal{E}_{un}(d_j, q^3) = \phi_{un}^2(MIN_{un}((c_1^2, 0), \mathcal{E}_{un}(d_j, q_1^1)),$$

$$\ldots, MIN_{un}((c_\eta^2, 0), \mathcal{E}_{un}(d_j, q_\eta^1))),$$

siendo η el número de átomos de q^3.

4. *Consulta expresada en CNF*:

$$\mathcal{E}_{un}(d_j, q^4) = \phi_{un}^1(\mathcal{E}_{un}(d_j, q_1^3), \ldots, \mathcal{E}_{un}(d_j, q_\omega^3)),$$

siendo ω el número de subexpresiones disyuntivas.

5. *Consulta expresada en DNF*:

$$\mathcal{E}_{un}(d_j, q^5) = \phi^1_{un}(\mathcal{E}_{un}(d_j, q^2_1), \ldots, \mathcal{E}_{un}(d_j, q^2_\omega)),$$

siendo ω el número de subexpresiones conjuntivas.

Por último, el resultado del sistema para cualquier consulta q proporcionada por el usuario es un conjunto difuso de documentos caracterizado por la función lingüística de pertenencia $\mathcal{E}_{un}$:

$$\{(d_1, \mathcal{E}_{un}(d_1, q^k)), \ldots, (d_m, \mathcal{E}_{un}(d_m, q^k))\}, k \in \{1, 2, 3, 4, 5\}.$$

Los documentos son mostrados en orden decreciente de $\mathcal{E}$ y dispuestos en clases lingüísticas de relevancia, donde el número de clases está limitado por la cardinalidad del conjunto no balanceado de etiquetas elegido para representar la variable lingüística *Relevancia*.

5.3. Ejemplo Teórico del Rendimiento del Nuevo Sistema de Recuperación de Información Lingüístico No Balanceado Definido

En esta Sección mostramos un ejemplo de rendimiento del SRI propuesto haciendo uso de información lingüística no balanceada.

Supongamos una pequeña base de datos con un conjunto de siete documentos $\mathcal{D} = \{d_1, \ldots, d_7\}$, representado por medio de un conjunto de diez términos $\mathcal{T} = \{t_1, \ldots, t_{10}\}$. Los documentos son indexados por medio de una función de indexación numérica $\mathcal{F}$, cuyas representaciones son:

$d_1 = 0.7/t_5 + 0.4/t_6 + 1/t_7$

$d_2 = 1/t_4 + 0.6/t_5 + 0.8/t_6 + 0.9/t_7$

$d_3 = 0.5/t_2 + 1/t_3 + 0.8/t_4$

$d_4 = 0.9/t_4 + 0.5/t_6 + 1/t_7$

$d_5 = 0.7/t_3 + 1/t_4 + 0.4/t_5 + 0.8/t_9 + 0.6/t_{10}$

$d_6 = 0.8/t_5 + 0.99/t_6 + 0.8/t_7$

$d_7 = 0.8/t_5 + 0.02/t_6 + 0.8/t_7 + 0.9/t_8$

Asumimos también el conjunto de siete etiquetas no balanceadas dadas en el ejemplo de la Figura 5.1 y su jerarquía lingüística mostrada en la Figura 5.2, la cual tiene dos niveles: $LH = l(1,5) \bigcup l(2,9)$, donde:

- $l(1,5) = \{N, L, M, H, T\}$ y
- $l(2,9) = \{N, VL, QL, L, M, H, QH, VH, T\}$

Con esto, representamos estos documentos usando el modelo de representación de información lingüística 2-tupla aplicando la función Δ sobre los términos índice ponderados $\mathcal{F}(d_j, t_i)$ y una transformación $\tau_t^{t'}$, $t \in \{t^-, t^+\}$, donde $t^- = l(1,5)$ y $t^+ = l(2,9)$:

$d_1 = (QH, -.4)/t_5 + (M, -.4)/t_6 + (T, .0)/t_7$

$d_2 = (T, .0)/t_4 + (H, -.2)/t_5 + (QH, .4)/t_6 + (VH, .2)/t_7$

$d_3 = (M, .0)/t_2 + (T, .0)/t_3 + (QH, .4)/t_4$

$d_4 = (VH, .2)/t_4 + (M, .0)/t_6 + (T, .0)/t_7$

$d_5 = (QH, -.4)/t_3 + (T, .0)/t_4 + (M, -.4)/t_5 + (QH, .4)/t_9 + (H, -.2)/t_{10}$

$d_6 = (QH, .4)/t_5 + (T, -.08)/t_6 + (QH, .4)/t_7$

$d_7 = (QH, .4)/t_5 + (N, .08)/t_6 + (QH, .4)/t_7 + (VH, .2)/t_8$

Si un usuario formula la consulta: $q = ((t_5, QH, VH) \wedge (t_6, L, L)) \vee (t_7, H, L)$ el sistema la evalúa como sigue:

- *Preprocesamiento de la consulta:* La consulta q está en DNF, pero tiene una subexpresión con solo un átomo, por tanto, q debe ser preprocesada y transformada en otra expresión equivalente donde cada subexpresión tenga al menos dos átomos. Entonces, q es transformada en $q^{'} = ((t_5, QH, VH) \vee (t_7, H, L)) \wedge ((t_6, L, L) \vee (t_7, H, L))$, y que ahora está expresada en CNF.

- *Evaluación de los átomos con respecto a la semántica de umbral:* Después de que la consulta q se ha transformado en $q^{'}$, evaluamos los átomos con respecto a la semántica de umbral por medio de g_{un} y obtenemos:

  - t_5:

  $$\{RSV_6^{5,1} = (QH, .4), RSV_7^{5,1} = (QH, .4)\}$$

  - t_6:

  $$\{RSV_1^{6,1} = (M, -.4), RSV_2^{6,1} = (QH, .4),$$
  $$RSV_4^{6,1} = (M, .0), RSV_6^{6,1} = (T, -.08)\}$$

  - t_7:

  $$\{RSV_1^{7,1} = (T, .0), RSV_2^{7,1} = (VH, .2), RSV_4^{7,1} = (T, .0),$$
  $$RSV_6^{7,1} = (QH, .4), RSV_7^{7,1} = (QH, .4)\}$$

  donde, por ejemplo $RSV_2^{7,1}$ se calcula como sigue:

  $$RSV_2^{7,1} = g_{un}(d_2, t_7, H) = (VH, .2),$$

  such that $(VH, .2) \geq (H, .0)$ and $(VH, .2) = \Delta(8 \cdot 0.9)$, with $8 = n(t^+) - 1$ and $0.9 = F(d_2, t_7)$.

- *Evaluación of subexpresiones y modelado de la semántica de importancia relativa:* La consulta $q^{'}$ tiene dos subexpresiones y ambas presentan dos átomos, $q_1^{'} =$

$(t_5, QH, VH) \vee (t_7, H, L)$ y $q_2^{'} = (t_6, L, L) \vee (t_7, H, L)$. Cada subexpresión está en forma disyuntiva, y por tanto, debemos de usar el operador no balanceado, ϕ_{un}^2 con un $orness(W) > .5$ (por ejemplo, con $(W = [.8, .2])$) junto con la función de transformación $MIN_{un}(weight$,valor no balanceado) para evaluarlas. Con esto, obtenemos los siguientes resultados:

- $q_1^{'}$:

$$\{RSV_1^1 = (L, -.2), RSV_2^1 = (L, -.2), RSV_4^1 = (L, -.2),$$

$$RSV_6^1 = (QH, -.48), RSV_7^1 = (QH, -.48)\}$$

- $q_2^{'}$:

$$\{RSV_1^2 = (L, .0), RSV_2^2 = (L, .0), RSV_4^2 = (L, .0),$$

$$RSV_6^2 = (L, .0), RSV_7^2 = (L, -.2)\}$$

donde, por ejemplo RSV_6^1 se calcula así:

$$RSV_6^1 = \phi_{un}^2(MIN_{un}((c_5^2, 0), RSV_6^{5,1}),$$

$$MIN_{un}((c_7^2, 0), RSV_6^{7,1})) =$$

$$= \phi_{un}^2(MIN_{un}((VH, 0), (QH, .4)),$$

$$MIN_{un}(L, 0), (QH, .4))) = \phi_{un}^2((QH, .4), (L, .0))$$

$$= \Delta^{-1}(\tau_{t'}^{t^+}(QH, .4)) \cdot 0.8 + \Delta^{-1}(\tau_{t'}^{t^-}(L, .0)) \cdot 0.2$$

$$= \Delta^{-1}(QH, .4) \cdot 0.8 + \Delta^{-1}(QL, .0) \cdot 0.2 = \Delta(5.52)$$

$$= (QH, -.48) \Longrightarrow \tau_{t^+}^{t^{'}}(QH, -.48) = (QH, -.48),$$

such that $t^+ = t^{'}$.

- *Evaluación de la consulta completa:* Obtenemos la evaluación de la consulta completa haciendo uso del operador no balanceado ϕ^1_{un} con $orness(W) < .5$ (po ejemplo $(W = [.2, .8])$).

$$\{RSV_6 = (L, .352), RSV_7 = (L, .192), RSV_1 = (L, -.16),$$
$$RSV_2 = (L, -.16), RSV_4 = (L, -.16)\}.$$

El mejor documento recuperado es d_6, que ha sido calculado como:

$$RSV_6 = \phi^2_{un}(RSV^1_6, RSV^2_6)$$

$$= \Delta^{-1}(\tau^{t^+}_{t'}(QH, -.48)) \cdot 0.2 + \Delta^{-1}(\tau^{t^-}_{t'}(L, .0)) \cdot 0.8$$

$$= \Delta^{-1}(QH, -.48) \cdot 0.2 + \Delta^{-1}(QL, .0) \cdot 0.8 = \Delta(2.704)$$

$$= (L, -.296) \Longrightarrow \tau^{t'}_{t^-}(L, -.296) = (L, .352).$$

5.4. Ejemplo Práctico del Rendimiento del Nuevo Sistema de Recuperación de Información Lingüístico No Balanceado Definido

En este apartado se muestran los resultados obtenidos por SRI_{un} sobre las mismas consultas realizadas en los apartados 4.3.3 y 4.4.2, pero utilizando un conjunto no balanceado de siete etiquetas $\mathcal{S}_{un} = \{N, L, M, H, VH, EH, T\}$. Para representar este conjunto de etiquetas es necesario usar los niveles 2 y 3 de la jerarquía lingüística de la Figura 3.9. Los resultados son mostrados en las Tablas 5.1 y 5.2.

Para más ejemplos véase el Anexo B, en el cual se podrán encontrar también los documentos en los que aparecen los términos usados en los siguientes ejemplos.

	RESULTADOS DE LA CONSULTA	
	Parametros utilizados	
Jerarquia = (2,3)	*Base de Datos* = "trec"(5000 Docs.)	*orness* = _
	Numero de documentos recuperados = 13	
Rank	**ID Doc**	**RSV**
1#	4220	(L,-0.50)
2#	3030	(L,-0.50)
3#	4133	(N,0.40)
4#	4782	(N,0.33)
5#	4157	(N,0.30)
6#	4459	(N,0.30)
7#	2621	(N,0.28)
8#	4097	(N,0.26)
9#	4984	(N,0.26)
10#	185	(N,0.25)
11#	1816	(N,0.22)
12#	2423	(N,0.22)
13#	1980	(N,0.21)

Tabla 5.1: Evaluación de $\langle clamp, H, L, \_\rangle$ con SRI_{un}.

	RESULTADOS DE LA CONSULTA	
	Parametros utilizados	
Jerarquia = (2,3)	*Base de Datos* = "trec"(5000 Docs.)	*orness* = 1.00
	Numero de documentos recuperados = 2	
Rank	**ID Doc**	**RSV**
1#	185	(N,0.16)
2#	2423	(N,0.14)

Tabla 5.2: Evaluación de $\langle bay, H, L, \_\rangle AND \langle clamp, T, L, \_\rangle$ con SRI_{un}.

Capítulo 6

Comentarios Finales

A continuación resaltaremos las conclusiones del trabajo realizado en la presente memoria y concluiremos la misma presentando las líneas de trabajo futuro.

6.1. Conclusiones

A lo largo de esta memoria nuestro objetivo ha sido, por un lado, el de presentar técnicas que flexibilicen la formulación de las necesidades de información del usuario; y por otro, el de mejorar algunos de los procesos internos de los SRIs que ocurren en la fase de recuperación. Estos dos subobjetivos a su vez, convergen en uno mucho más ambicioso consistente en mejorar la satisfacción del usuario, a través de una interacción mucho más flexible entre él y el sistema.

Para lograr esto hemos tenido que profundizar en el área del modelado de sistemas de recuperación de información documental desde el punto de vista del modelado lingüístico difuso.

Para tal propósito, se han desarrollado distintas aproximaciones, cada una de las

cuales aborda aspectos concretos de la recuperación y en conjunto dan lugar a un meta-modelo de sistema de recuperación de información que servirá de marco general de aplicación, manteniendo además requisitos de flexibilidad de expresión, facilidad de uso, e interpretabilidad de los resultados obtenidos, facilitando en última instancia el diseño y uso de estos sistemas.

Más concretamente, la labor realizada puede resumirse en:

1. Hemos desarrollado un nuevo modelo lingüístico 2-tupla de SRI Documental que aportar a los anteriormente propuestos lo siguiente:

   a) una mejora en la interpretación de la semántica de umbral simétrico,

   b) una mejora en los procesos de agregación, dando como resultado operadores de agregación mucho más flexibles, y

   c) una mejora global del proceso de recuperación, gracias al uso del modelo de representación de información lingüística 2-tupla, junto con las mejoras anteriores.

   Este sistema obtiene resultados más interpretables, eliminando la pérdida de información y precisión que se producía en los anteriores modelos.

2. Hemos desarrollado un segundo modelo lingüístico de SRI Documental que permite usar información lingüística no balanceada, y que permite al usuario flexibilizar aún más la expresión de sus necesidades de información.

3. Hemos evaluado los distintos modelos propuestos, comparándolos con otros propuestos en la literatura. Con esta evaluación hemos constatado que, efectivamente los modelos lingüísticos de SRI propuestos mejoran notablemente los juicios de relevancia de la evaluación final de los documentos, lo que suponemos mejorará la satisfacción del usuario.

4. Hemos implementado todos los desarrollos teóricos descritos en esta memoria de tesis, en un software que permitirá el desarrollo de futuros estudios, así como de un entorno de evaluación.

6.2. Trabajos Futuros

Algunos de los trabajos que abordaremos en el futuro son:

- Estudiar las diferentes funciones de evaluación de consultas existentes en la literatura, con el objeto de definir un marco general de aplicación que facilite el diseño y el uso de los sistemas de recuperación de información.

- Diseñar herramientas híbridas de soft computing (algoritmos genéticos - modelado lingüístico difuso 2-tupla) para asistir al usuario a definir, de manera precisa y flexible, sus necesidades de información partiendo de trabajos desarrollados y del software desarrollado en esta memoria.

- Desarrollo de un interfaz gráfica, amigable y flexible que asista al usuario en el complicado proceso de formulación de sus consultas.

- Desarrollo de un modelo de evaluación, que nos permita valorar el alcance real de las mejoras introducidas, principalmente en el ámbito cognitivo, que nos permita obtener el valor de satisfacción real del usuario.

- Aplicar las mejoras desarrolladas sobre los SRI lingüísticos, presentadas en la presente memoria, al ámbito Web.

Apéndice A

Implementación de los Nuevos Modelos de Sistemas de Recuperación de Información Lingüísticos Propuestos

En este anexo se describirán los apartados más relevantes de la implementación del SRI Difuso Lingüístico 2-tupla completo, desarrollado en esta memoria de tesis. Para ello partiremos comentando el esquema en diagrama de bloques del sistema y posteriormente, se detallará las apartados más destacados.

El sistema de recuperación de información desarrollado se divide fundamentalmente en tres bloques: *lenguaje de consulta*, *subsistema de evaluación* y *subsistema de almacenamiento*. El sistema se puede visualizar gráficamente en la Figura A.1.

A.1. Lenguaje de Consulta. Implementación.

El lenguaje de consulta es la herramienta por la cual el usuario puede indicar al sistema las preferencias de información que necesita. En este sentido, debe ser lo suficientemente expresivo y a la vez sencillo de usar. En la Sección *modelo difuso lingüístico*

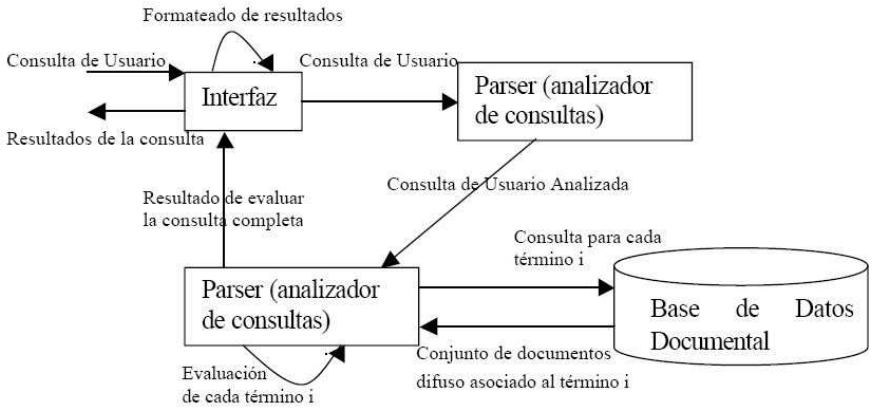

Figura A.1: Diagrama de bloques del sistema

ordinal del Capítulo 4 se especifica formalmente su sintaxis, aquí utilizaremos otra notación, gramática establecida mediante producciones. Esta nueva visión del lenguaje nos permitirá utilizar, a su vez, otra herramienta para realizar el análisis sintáctico de las consultas introducidas por el usuario. Esta herramienta es YACC que posteriormente se analizará.

Utilizando notación BNF el lenguaje de consulta se puede definir de la siguiente manera:

```
-t (NIVEL_JERARQUIA_IZDA, NIVEL_JERARQUIA_DCHA) -d DATABASE -q
<consulta> -o ORNESS

<consulta>::= <atomos>
            | OPERADOR <atomos>
            | FN <subexpresiones>
```

```
<atomo>::= NOT <TERMINO, ETIQUETA, ETIQUETA, ETIQUETA>
           | <TERMINO, ETIQUETA, ETIQUETA, ETIQUETA>

<atomos>::= <atomos> <atomo>
            | <atomo>

<subexpresion>::= ( <atomos> )
                  | ( ETIQUETA <atomos> )

<subexpresiones>::= <subexpresiones> <subexpresion>
                    | <subexpresion>
```

Donde `OPERADOR` puede ser uno de los operadores lógicos (AND, OR), `FN` indica si la consulta está expresada en forma normal, donde los valores posibles serán: FNC (forma normal conjuntiva) y FND (forma normal disyuntiva); `DATABASE` es el nombre de la colección utilizada, `TERMINO` representará a cualquiera de los términos de indización usados en `DATABASE`, por su parte, `ORNESS` establece el valor de la medida orness del operador de agregación LOWA utilizado para evaluar la consulta completa[1] , y finalmente, `NIVEL_JERARQUIA_IDZA` e `NIVEL_JERARQUIA_DCHA` indican los niveles de la jerarquía lingüística LH usados para representar las etiquetas lingüísticas de los lados izquierdo y derecho de la etiqueta central, respectivamente. Los únicos valores posibles serán: 1 (para el nivel menos granular de LH y etiquetas $\{N, M, T\}$), 2 (etiquetas $\{N, L, M, H, T\}$) y 3 (etiquetas $\{N, EL, VL, L, M, H, VH, EH, T\}$).

[1] Por simetría, la medida orness del operador LOWA ponderado que evalúa las subexpresiones se obtiene como 1-`ORNESS`.

Con esta gramática es posible gestionar todas las expresiones o consultas soportadas por los dos modelos lingüísticos ordinales propuestos en [50, 51] y por los dos nuevos modelos de SRI propuestos en esta memoria de tesis.

Para su implantación en el sistema, se ha usado la herramienta de análisis sintáctico y semántico por excelencia, YACC. YACC lee de un fichero la definición de la gramática, y a partir de ésta construye la máquina de estados capaz de analizar las consultas descritas con esa gramática. Al utilizar YACC conseguimos varias cosas: la primera, nos evitamos el tedioso trabajo de construir una máquina nosotros mismos, con lo que ello conlleva en cuanto a tiempo y probabilidad de error; segundo, utilizamos una herramienta estándar, con ello, cualquier persona que desee modificar el lenguaje de consulta, puedo hacerlo de una manera sencilla.

En el Anexo B mostraremos algunos ejemplos de estas consultas, junto con los resultados obtenidos por los SRI lingüísticos propuestos en esta memoria.

A.2. Subsistema de Evaluación. Implementación.

Cuando el analizador de consultas recibe una, la procesa y las distintas secciones que detecta (operadores, términos, subexpresiones) las va introduciendo en la estructura mostrada en la Figura A.2.

Esta estructura es capaz de almacenar las distintas subexpresiones que se pueden formar. En ella podemos apreciar una estructura alborea, donde las ramas hoja (en la figura son las que aparecen en vertical) almacena toda la información asociada a un término. Esta información consiste de: *identificador de término*, *pesos* para las tres semánticas (umbral simétrico, cuantitativa e importancia relativa). Adicionalmente,

también es capaz de albergar los resultados parciales de las respectivas evaluaciones: con respecto a umbral y cuantitativa. Por su parte, en horizontal, podemos ver una lista de subexpresiones (nodos intermedios), éstas almacenan la lista de átomos que le corresponden y además, disponen de un campo *peso*, que sólo tiene sentido cuando hablamos de la segunda variante del modelo se SRI lingüístico ordinal propuesto por Herrera-Viedma en [50]; y de un campo para almacenar el resultado de la evaluación de la semántica de importancia relativa a nivel de términos (para los dos modelos de SRI lingüísticos propuestos en esta memoria y el modelo de SRI lingüístico ordinal propuesto por Herrera-Viedma en [51]). Obviamente, ambos campos extra no se utilizan simultáneamente.

Una vez que el analizador ha reconocido la consulta y la ha estructurado de la manera comentada, es el momento que entre en acción el módulo de evaluación. Como se comento en los Capítulos 4 y 5, el subsistema de evaluación procede de una forma ascendente, evaluando en una primera etapa los átomos (evaluando las semánticas de umbral y cuantitativa), continuando en una segundo lugar con las subexpresiones, combinando los resultados parciales de la evaluación de los átomos; y por último, agregando los resultados de las distintas subexpresiones, terminando en ese momento con la evaluación completa de la consulta.

A.2.1. ¿Por Qué Esta Representación?

Una de las características de la evaluación en estos modelos, es que las consultas se presenten en forma normal (FNC o FND), con las restricciones específicas de cada modelo. Por generalidad, podemos hacer un SRI que acepte cualquier tipo de consultas,

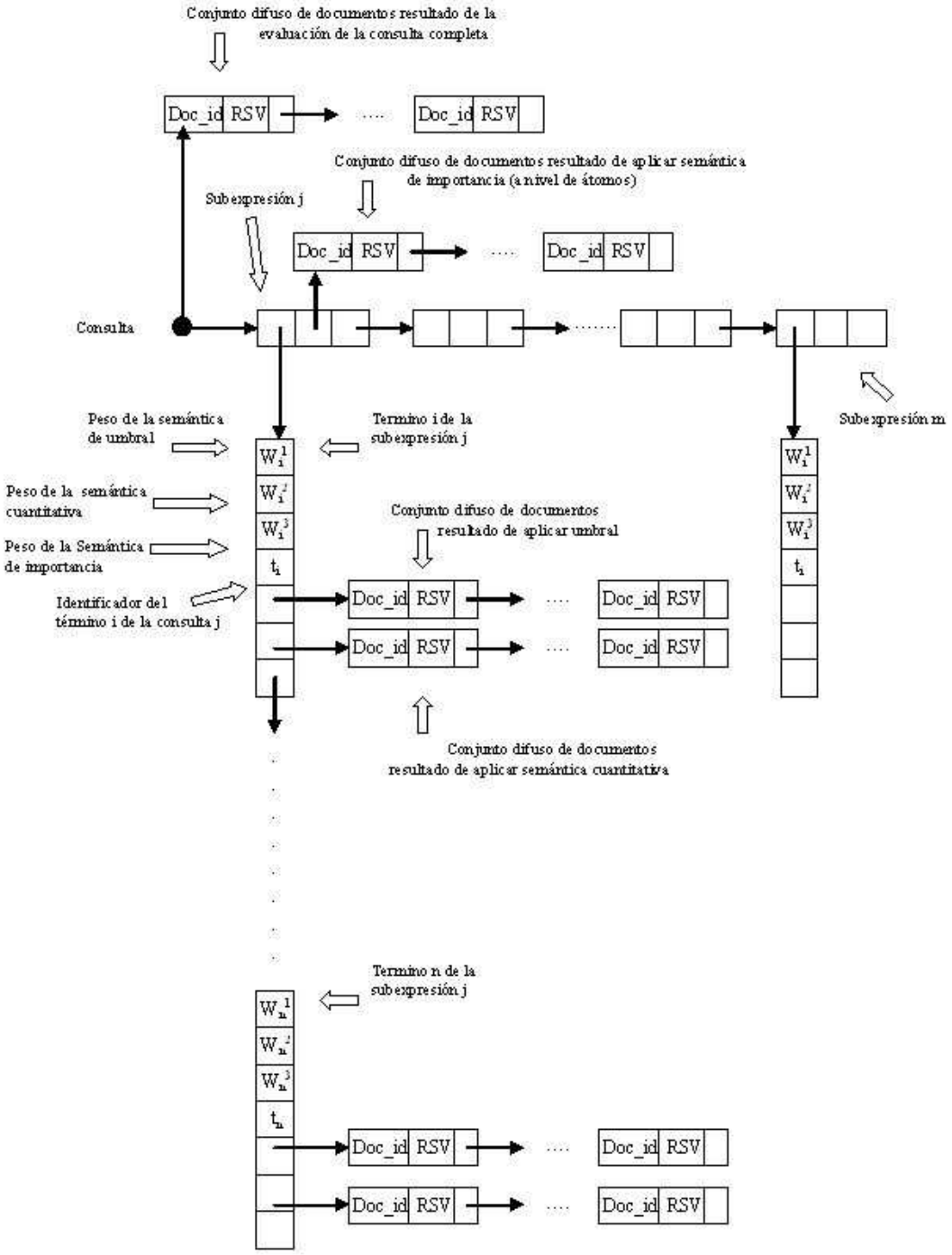

Figura A.2: Estructura de datos para representar las consultas

estén éstas en forma normal o no. Si no están en forma normal, es necesario un preprocesamiento previo. Un ejemplo de este proceso lo podemos ver mediante la siguiente consulta:

$$q = (t_2\ OR\ t_6\ OR\ t_4)\ AND\ t_7$$

si la normalizamos obtenemos:

$$q^{'} = (t_2\ AND\ t_7)\ OR\ (t_6\ AND\ t_7)\ OR\ (t_4\ AND\ t_7)$$

Si observamos en detalle, vemos que el término t_7 aparece tres ves (una vez en cada subexpresión). Si utilizáramos un proceso de evaluación ascendente clásico, nos veríamos obligados a evaluar tres veces el término t_7, con lo que ello conlleva (acceso a base de datos, recuperación de documentos relevantes, aplicación de umbral, aplicación de semántica cuantitativa si procede, etc). Supongamos, por ejemplo, lo que supondría evaluar una consulta proporcionada como resultado por las técnicas IQBE comentadas en la Sección 2.5.

Por estos motivos, en este trabajo se propone un enfoque alternativo al de la evaluación ascendente clásica. El nuevo esquema de evaluación procede como se muestra en el Algoritmo A.1.

Para cada una de las subexpresiones y para cada uno de sus átomos $a_i = \langle t_j, c_j^1, c_j^2, c_j^3 \rangle$

Si a_i no ha sido evaluado con respecto a semántica de umbral, evaluarlo.

1. Evaluar átomo con respecto a semántica cuantitativa.
2. Establecer átomo como evaluado.
3. Si en alguna otra subexpresión existe un átomo a_k con el mismo identificador t_j de término y coinciden ambos en su ponderación umbral:
   - *a*) hacer resultado de evaluar átomo a_k con respecto a la semántica de umbral igual al resultado de evaluar a_i con respecto al umbral.
   - *b*) establecer a_k como evaluado con respecto a la semántica de umbral para comprobaciones posteriores.
   - *c*) Si además coinciden ambos en su ponderación cuantitativa:
     1) Hacer: resultado de evaluar a_k con respecto a semántica cuantitativa igual al resultado de evaluar a_i cuantitativamente.
     2) Establecer a_k como evaluado.

Algoritmo A.1: Algoritmo de evaluación átomos. Aplicación del criterio de separabilidad.

Como podemos apreciar en el Algoritmo A.1, éste nos evita el tener que evaluar un

mismo término más de una vez. Para que esto sea posible, los átomos con el mismo identificador de término deben coincidir al menos en su ponderación umbral, hecho que se satisface en la normalización de las consultas.

Como característica adicional, y derivada de su manera de actuar, podemos notar que a medida que vamos evaluando las expresiones (bucle externo), el número de evaluaciones de átomos a realizar desciende.

Una vez evaluados los átomos, el siguiente paso consiste en evaluar la subexpresión asociada, esto no es más que aplicar el operador adecuando. Este paso, nos produce el conjunto difuso de documentos asociados a la subexpresión, que después serán agregados. Como nota adicional, podemos decir que la estructura desarrollada proporciona una posibilidad de expansión adicional.

Ésta consiste en incorporar la semántica cuantitativa a la consulta completa Esta ponderación cuantitativa nos limitaría el número de documentos devueltos por el SRI, es decir, nos establecería el número máximo de documentos del ranking. Por supuesto, vendría dada mediante valores lingüísticos y se evaluaría también con el mismo algoritmo que el utilizado para evaluar la semántica cuantitativa a nivel de términos.

A.3. Representación de los Documentos. Base de Datos.

Como ya hemos comentado, internamente trabajamos contra una base de datos, en esta implementación particular hemos utilizado *PostgreSQL*. En esta base de datos los documentos están almacenados en dos tablas por colección. Una tabla sería el *diccionario*, es decir donde están los términos, entendiendo éstos como palabras que podemos encontrar en los documentos y el *término de indexación* asociado a cada uno.

Como hemos usado SMART para el proceso de la indexación, podremos comprobar cómo algunas palabras tienen asociado el mismo término de indexación. Esto es debido a que SMART incorpora un sistema de extracción de raíces o Stemming lo cual produce que por ejemplo dos palabras con la misma raíz, sean consideradas la misma para el sistema de evaluación. La otra tabla contiene para cada pareja (*término de indexación*, *documento*) un *peso* asociado que determina la relevancia del término en el documento (sería una especie de fichero invertido).

A.3.1. Utilizando SMART como Indexador

Para evitar tener que construir un indexador completo para poder hacer los experimentos, se ha optado por la opción de usar colecciones indexadas por el sistema SMART. A tal efecto se ha considerado interesante introducir en esta documentación una breve descripción de este sistema.

Basado en el modelo del espacio vectorial, el SRI SMART [18, 82, 83] ha sido y sigue

siendo referencia para multitud de investigadores en todo el mundo. Fue desarrollado en los años sesenta en la Universidad de Cornell por un grupo de investigadores dirigidos por G. Salton y está disponible en la dirección ftp.cs.cornell.edu del Departamento de Ciencias de la Computación de la citada universidad.

SMART está formado por un conjunto de programas que componen un sistema completo de recuperación automática de documentos. Permite la creación, mantenimiento y uso de colecciones de documentos, de tamaños pequeños a medios. Pero ante todo, sus autores lo definen como una herramienta experimental para investigar métodos y técnicas de RI. Además de crear una herramienta flexible, apta para la experimentación, los desarrolladores no se olvidaron de los usuarios, por lo que elaboraron un entorno rápido, portable e interactivo (a pesar de no poseer una interfaz de usuario gráfica).

Este SRI está compuesto por cuatro módulos básicos:

- Módulo de indexación: convierte cualquier colección de documentos en su formato original a vectores de términos.

- Módulo de recuperación: calcula la similitud, basada en la función del coseno (expresión (1.6)), entre los documentos y una consulta, ya indexados previamente, generando como resultado una lista ordenada decrecientemente por dicha similitud de todos los documentos de la colección que es mostrada al usuario. Además, implanta una organización de los documentos con contenidos próximos en grupos, facilitando así la posterior recuperación.

- Módulo de realimentación de relevancia: a partir de los resultados de una consulta

ya formulada y de los juicios de relevancia expresados por el usuario, genera una nueva consulta para recuperar más documentos relevantes.

- Módulo de evaluación, a partir de dicha lista ordenada y de los juicios de relevancia establecidos en las colecciones de prueba, genera las curvas de exhaustividad-precisión.

Al estar desarrollado en el lenguaje C y al distribuirse su código gratuitamente, permite utilizar todas las rutinas desarrolladas para implantar estos cuatro módulos anteriores, de tal forma que desde un programa externo se puede acceder a la información que almacena este SRI sin necesidad de utilizarlo como medio para ello.

Seguidamente vamos a esbozar el mecanismo que aplica SMART a la hora de calcular el peso de los términos de los documentos y las consultas, ya que se reseñará posteriormente en el desarrollo de esta memoria. El proceso de generación de los pesos de cada término de un vector se compone de tres fases, que parten de la frecuencia del término en el documento o consulta (tf) [81]:

1. Normalización del tf de cada término del vector.
2. Modificación del peso calculado en la etapa anterior, generalmente con información proveniente de la colección completa, para así aumentar el peso de los términos menos comunes y disminuir el de los más comunes.
3. Normalización del vector completo.

El proceso completo de ponderación se nota mediante una palabra de tres caracteres. Cada uno de ellos representa el método empleado en la fase correspondiente, existiendo

un total de cinco alternativas posibles para cada uno. Veamos como ejemplo la interpretación de los esquemas de ponderación utilizados a lo largo de los experimentos hechos con SMART en el desarrollo de esta memoria:

- Esquema *nnn*:

  1. n $\rightarrow$ No se hace ninguna conversión, dejando el valor del tf intacto.
  2. n $\rightarrow$ No se combina el tf con ninguna información de la colección.
  3. n $\rightarrow$ No se normaliza el vector completo.

  Quedando al final como peso en todos los términos el tf de cada uno.

- Esquema *ntn*

  1. n $\rightarrow$ No se hace ninguna conversión, dejando el valor del tf intacto.
  2. t $\rightarrow$ Se calcula el peso $tf \cdot idf$ del término.
  3. n $\rightarrow$ No se normaliza el vector completo.

- Esquema *ntc*

  1. n $\rightarrow$ No se hace ninguna conversión, dejando el valor del tf intacto.
  2. t $\rightarrow$ Se calcula el peso $tf \cdot idf$ del término.
  3. c $\rightarrow$ Se normaliza el vector completo por la raíz cuadrada de la suma de los $tf \cdot idf$ al cuadrado.

  El peso final será el $tf \cdot idf$ normalizado según la medida del *coseno*.

A.3.2. Colecciones Estándar de Prueba

A la hora de medir la calidad recuperadora de un SRI nos encontramos con el problema de que los documentos relevantes a una consulta son totalmente desconocidos y, por tanto, no se pueden determinar exactamente las curvas de exhaustividad y precisión. Los SRI experimentales deben, de alguna forma, tener ese conocimiento para conseguir una evolución positiva en su comportamiento. Por esto se suelen utilizar unas colecciones documentales de prueba, que constan de:

- Un conjunto de documentos, que contienen información como el título, autor, fecha y resumen, incluso las citas de unos a otros documentos.

- Un conjunto de consultas, efectuadas en lenguaje natural o en alguno formal, como puede ser el booleano.

- Un conjunto de juicios de relevancia para cada consulta, es decir, el conjunto de documentos que se consideran relevantes para todas las consultas contenidas en el segundo conjunto.

A pesar de ser una práctica ampliamente utilizada la de evaluar los SRI aplicándolos a colecciones de prueba, Turtle argumenta como inconvenientes [89] que no poseen un tamaño acorde con el que tienen las colecciones reales, que las representaciones de los documentos se obtienen de resúmenes y no del texto completo, y que los juicios de relevancia se ven afectados por una gran cantidad de factores. En esta memoria presentamos resultados de los experimentos realizados con una de estas colecciones, TREC.

TREC

Los investigadores en recuperación información han sido frecuentemente criticados desde dos frentes. El primero es la falta de un sistema formal sólido como fundamento base. El segundo es la falta de robustos y consistentes tests y benchmarks. Durante tres décadas una experimentación en recuperación información estuvo basada en relativamente pequeñas colecciones las cuales no reflejaron el principal tema presente en un extenso en torno bibliográfico. A principios de los 90, como reacción a esta desorganización y bajo el liderazgo de Donna Harman del Instituto Nacional de Estándares y Tecnología (NIST) en Maryland se promocionó una serie de conferencias anuales dedicadas a la experimentación con grandes colecciones de documentos en la que comprendiendo más de millones de documentos. Tales conferencias recibieron el nombre de TREC por Text REtrieval Conference. Los encuentros de investigación que participaron en las conferencias usaron de referencia aquellos experimentos para comparar sus sistemas de recuperación.

Las series de conferencias TREC fueron patrocinadas por el NIST y por la Oficina de Tecnologías de la Información de la Defense Advanced Research Project Agency (DARPA) como parte del programa TIPSTER. Aunque la colección fue creada bajo el programa TIPSTER, es frecuentemente referida como TIPSTER o la colección de test TIPSTER/TREC. Nosotros, por simplicidad la llamamos TREC.

En concreto la versión de las conferencias TREC que hemos usado en esta memoria contienen un total de 5000 documentos.

CACM

La colección CACM consiste de 3204 artículos publicados en Communications of the ACM desde el primer ejemplar en 1958 hasta el último número en 1979. Estos documentos comprenden una vasta literatura de informática debido al hecho de que CACM fue por años la publicación número uno en el campo. La colección incluye junto al texto de los artículos información sobre los mismos estructurada en campos (llamados conceptos por Fox) que son los siguientes: nombres de los autores, fecha de la información, palabra provenientes de las secciones título y resumen (abstract), categorías derivadas de un esquema de clasificación jerárquico, referencias directas entre artículos, conectores de parejas bibliográficas, número de "co-referencias" para cada par de artículos (fuente y destino).

Los campos nombre del autor y fecha de información dan información sobre los autores y fecha de la publicación. El campo palabras provenientes nunca para cada documento una lista de los términos de indexación (de las secciones título y resumen) a las cuales se les ha aplicado una técnica de stemming para quedarnos sólo con las raíces de las palabras.

El campo categorías asigna una lista de categorías (del esquema Computing Review) de clasificación para cada documento. Puesto que las categorías son bastante generales, el número de categorías para cualquier documento suelen ser normalmente menores de cinco. El campo referencias directas da una lista de parejas de documentos [da,db], según la cual para cada documento da existe una referencia directa hacia el documento db. El campo conectores de parejas bibliográficas da una lista de tripletas $[d_1, d_2, n\_cited]$ en la cual los documentos d_1 y d_2 incluyen ambos una referencia al mis-

mo tercer documento d_j y el factor n_cited cuenta el número de documentos citados por ambos. El campo "co-referencias" ofrece una lista de tripletas $[d_1, d_2, n\_citing]$ en los cuales tanto el documento d_1 como d_2 son citados desde un tercer documento d_j que es el mismo. El factor n_citing cuenta el número de documentos d_j que hacen referencia a sendos documentos. Así la colección CACM ofrece un entorno extraordinario para probar algoritmos de recuperación basada en información derivada de de patrones de referencias cruzadas, un tema que ha llamado la atención mucho en el pasado.

Esta colección también incluye un conjunto de 52 consultas de test. Para cada petición de información la colección también incluyen dos consultas formuladas en forma booleana y un conjunto de documentos relevantes. Ya que las peticiones de información son bastante específicas, el número medio de documentos relevantes para cada petición información es pequeño, en torno a 15. Como resultado, en la precisión y el recall tienden a ser bajos con esta colección.

Apéndice B

Experimentación Práctica de los Nuevos Modelos de Sistemas de Recuperación de Información Lingüísticos Propuestos

En este anexo, mostramos ejemplos de rendimiento de los dos modelos de SRI lingüísticos propuestos en esta memoria de tesis: SRI lingüístico 2-tupla (SRI_{2t}) y SRI lingüístico no balanceado (SRI_{un}).

B.1. Representación de los Términos Utilizados en los Experimentos

En esta memoria hemos utilizado los siguientes términos: *clamp*, *bay*, *jordan*, *examin*. En las siguientes tablas pueden verse los documentos en los que estos términos aparecen así como el peso asignado por SMART.

DOCUMENTOS EN LOS APARECE *clamp*	13
ID Doc	**peso**
185	0.078764
1816	0.070199
1980	0.064766
2423	0.068272
2621	0.087286
3030	0.156319
4097	0.081695
4133	0.126372
4157	0.094336
4220	0.156862
4459	0.092580
4782	0.104293
4984	0.079775

Tabla B.1: Documentos en los aparece *clamp*.

DOCUMENTOS EN LOS APARECE *bay*	47
ID Doc	**peso**
92	0.071695
185	0.061758
187	0.121878
284	0.067983
297	0.046314
601	0.089294
1196	0.049971
1225	0.055802
1337	0.155427
1749	0.077640
1764	0.063371
1913	0.066352
1922	0.069654
2385	0.077320
2423	0.055591
2564	0.210819
2618	0.075224
2633	0.103426
2843	0.050127
2929	0.064795
2973	0.249027
3331	0.063544
3370	0.063345
3374	0.092531
3378	0.073091
3467	0.069758
3517	0.094365
3824	0.066491
3861	0.083889
3886	0.214138

Tabla B.2: Documentos en los aparece *bay*.

DOCUMENTOS EN LOS APARECE *bay*	47
ID Doc	**peso**
4057	0.154972
4265	0.153516
4355	0.070425
4391	0.243884
4493	0.050834
4528	0.072424
4558	0.152493
4569	0.075580
4606	0.063759
4669	0.282508
4703	0.105647
4720	0.125385
4724	0.269034
4733	0.144172
4742	0.055956
4955	0.044928
4964	0.076723

Tabla B.3: Documentos en los aparece *bay* (Continuación).

DOCUMENTOS EN LOS APARECE *examin*	143
ID Doc	**peso**
129	0.040927
150	0.098157
185	0.050170
235	0.044669
241	0.090340
251	0.118983
273	0.058566
381	0.080970
383	0.099467
422	0.066893
463	0.077807
486	0.141214
630	0.038139
640	0.038419
725	0.067244
756	0.091175
788	0.174591
802	0.053158
842	0.034254
854	0.054423
871	0.104756
958	0.089312
971	0.039452
993	0.049485
1026	0.060139
1060	0.078917
1063	0.115140
1082	0.067658
1108	0.064754
1124	0.074414

Tabla B.4: Documentos en los aparece *examin*.

DOCUMENTOS EN LOS APARECE *exain*	143
ID Doc	**peso**
1136	0.081237
1209	0.086482
1234	0.053975
1238	0.070367
1242	0.158034
1246	0.114440
1252	0.063089
1374	0.034345
1405	0.064918
1415	0.082619
1421	0.053635
1426	0.130167
1458	0.069615
1494	0.149385
1499	0.043268
1604	0.056776
1627	0.136066
1689	0.056617
1757	0.050046
1794	0.062371
1816	0.041920
1890	0.073789
1908	0.058677
1922	0.058100
2044	0.063408
2066	0.057192
2074	0.045571
2125	0.239098
2132	0.178497
2161	0.079676

Tabla B.5: Documentos en los aparece *examin* (Continuación I).

DOCUMENTOS EN LOS APARECE *examin*	47
ID Doc	**peso**
2269	0.058582
2271	0.069340
2291	0.090567
2327	0.054787
2342	0.069315
2364	0.045024
2374	0.056327
2380	0.140134
2395	0.061893
2422	0.071246
2423	0.042337
2445	0.058492
2454	0.045723
2458	0.120579
2466	0.103230
2467	0.055967
2497	0.149697
2509	0.104759
2521	0.106517
2609	0.089614
2694	0.091854
2760	0.042925
2782	0.225583
2840	0.067778
2866	0.098817
3027	0.107381
3096	0.207414
3130	0.066546
3136	0.051683
3193	0.045207

Tabla B.6: Documentos en los aparece *examin* (Continuación II).

DOCUMENTOS EN LOS APARECE *examin*	143
ID Doc	**peso**
3266	0.111072
3301	0.036569
3322	0.075245
3357	0.120438
3364	0.167132
3380	0.067875
3395	0.049646
3397	0.070745
3424	0.060942
3439	0.055046
3448	0.061053
3453	0.106632
3456	0.057057
3485	0.062455
3541	0.065976
3549	0.060232
3576	0.107030
3658	0.040499
3665	0.059923
3693	0.038794
3766	0.027838
3777	0.049240
3823	0.073516
3843	0.083987
3906	0.061754
3932	0.064459
3984	0.042687
4038	0.038021
4103	0.049955
4131	0.084856

Tabla B.7: Documentos en los aparece *examin* (Continuación III).

DOCUMENTOS EN LOS APARECE *examin*	143
ID Doc	**peso**
4172	0.055644
4204	0.099185
4216	0.125588
4299	0.059450
4301	0.074225
4364	0.082649
4399	0.106116
4464	0.046191
4553	0.062846
4621	0.081988
4706	0.091145
4727	0.079817
4776	0.155291
4777	0.037481
4863	0.050327
4865	0.131397
4885	0.091963
4913	0.053901
4940	0.136536
4968	0.076852
4977	0.054115
4985	0.032367
4986	0.068308

Tabla B.8: Documentos en los aparece *examin* (Continuación IV).

DOCUMENTOS EN LOS APARECE *jordan*	31
ID Doc	**peso**
185	0.076233
244	0.098079
305	0.112291
339	0.119972
432	0.103027
844	0.089385
989	0.089474
1312	0.098888
1384	0.061003
1421	0.076704
1621	0.075557
1724	0.093868
1784	0.065776
2023	0.131156
2031	0.044410
2078	0.082578
2120	0.227585
2525	0.095194
3131	0.072208
3142	0.084681
3308	0.070555
3312	0.070599
3325	0.097768
3766	0.044235
4148	0.146058
4234	0.069661
4288	0.075752
4325	0.055119
4518	0.064859
4542	0.045477
4745	0.114132

Tabla B.9: Documentos en los aparece *jordan*.

B.2. Más Ejemplos de Rendimiento con SRI_{2t}

	RESULTADOS DE LA CONSULTA	
	Parametros utilizados	
Jerarquia = (2,2)	*Base de Datos* = "trec"(5000 Docs.)	*orness* = _
	Numero de documentos recuperados = 143	
Rank	**ID Doc**	**RSV**
1#	2125	(EL,0.28)
2#	2782	(EL,0.20)
3#	3096	(EL,0.11)
4#	2132	(EL,-0.05)
5#	788	(EL,-0.07)
6#	3364	(EL,-0.11)
7#	1242	(EL,-0.16)
8#	4776	(EL,-0.17)
9#	2497	(EL,-0.20)
10#	1494	(EL,-0.20)
11#	486	(EL,-0.25)
12#	2380	(EL,-0.25)
13#	4940	(EL,-0.27)
14#	1627	(EL,-0.27)
15#	4865	(EL,-0.30)
16#	1426	(EL,-0.31)
17#	4216	(EL,-0.33)
18#	2458	(EL,-0.36)
19#	3357	(EL,-0.36)
20#	251	(EL,-0.37)
21#	1063	(EL,-0.39)
22#	1246	(EL,-0.39)
23#	3266	(EL,-0.41)
24#	3027	(EL,-0.43)
25#	3576	(EL,-0.43)

Tabla B.10: Evaluación de $\langle examin, VH, \_, \_ \rangle$ con SRI'_{2t}.

	RESULTADOS DE LA CONSULTA	
	Parametros utilizados	
Jerarquia = (2,2)	*Base de Datos* = "trec"(5000 Docs.)	*orness* = _
	Numero de documentos recuperados = 143	
Rank	**ID Doc**	**RSV**
26#	3453	(EL,-0.43)
27#	2521	(EL,-0.43)
28#	4399	(EL,-0.43)
29#	2509	(EL,-0.44)
30#	871	(EL,-0.44)
31#	2466	(EL,-0.45)
32#	383	(EL,-0.47)
33#	4204	(EL,-0.47)
34#	2866	(EL,-0.47)
35#	150	(EL,-0.48)
36#	4885	(N,0.49)
37#	2694	(N,0.49)
38#	756	(N,0.49)
39#	4706	(N,0.49)
40#	2291	(N,0.48)
41#	241	(N,0.48)
42#	2609	(N,0.48)
43#	958	(N,0.48)
44#	1209	(N,0.46)
45#	4131	(N,0.45)
46#	3843	(N,0.45)
47#	4364	(N,0.44)
48#	1415	(N,0.44)
49#	4621	(N,0.44)
50#	1136	(N,0.43)
51#	381	(N,0.43)
52#	4727	(N,0.43)
53#	2161	(N,0.42)
54#	1060	(N,0.42)
55#	463	(N,0.41)
56#	4968	(N,0.41)
57#	3322	(N,0.40)

Tabla B.11: Evaluación de $\langle examin, VH, \_, \_ \rangle$ con SRI'_{2t} (Continuación I).

	RESULTADOS DE LA CONSULTA	
	Parametros utilizados	
Jerarquia = (2,2)	*Base de Datos* = "trec"(5000 Docs.)	*orness* = _
	Numero de documentos recuperados = 143	
Rank	**ID Doc**	**RSV**
58#	1124	(N,0.40)
59#	4301	(N,0.40)
60#	1890	(N,0.39)
61#	3823	(N,0.39)
62#	2422	(N,0.38)
63#	3397	(N,0.38)
64#	1238	(N,0.38)
65#	1458	(N,0.37)
66#	2271	(N,0.37)
67#	2342	(N,0.37)
68#	4986	(N,0.36)
69#	3380	(N,0.36)
70#	2840	(N,0.36)
71#	1082	(N,0.36)
72#	725	(N,0.36)
73#	422	(N,0.36)
74#	3130	(N,0.35)
75#	3541	(N,0.35)
76#	1405	(N,0.35)
77#	1108	(N,0.35)
78#	3932	(N,0.34)
79#	2044	(N,0.34)
80#	1252	(N,0.34)
81#	4553	(N,0.34)
82#	3485	(N,0.33)
83#	1794	(N,0.33)
84#	2395	(N,0.33)
85#	3906	(N,0.33)
86#	3448	(N,0.33)

Tabla B.12: Evaluación de $\langle examin, VH, \_, \_ \rangle$ con SRI'_{2t} (Continuación II).

	RESULTADOS DE LA CONSULTA	
	Parametros utilizados	
Jerarquia = (3,3)	*Base de Datos* = "trec"(5000 Docs.)	*orness* = _
	Numero de documentos recuperados = 143	
Rank	**ID Doc**	**RSV**
87#	3424	(N,0.33)
88#	3549	(N,0.32)
89#	1026	(N,0.32)
90#	3665	(N,0.32)
91#	4299	(N,0.32)
92#	1908	(N,0.31)
93#	2269	(N,0.31)
94#	273	(N,0.31)
95#	2445	(N,0.31)
96#	1922	(N,0.31)
97#	2066	(N,0.31)
98#	3456	(N,0.30)
99#	1604	(N,0.30)
100#	1689	(N,0.30)
101#	2374	(N,0.30)
102#	2467	(N,0.30)
103#	4172	(N,0.30)
104#	3439	(N,0.29)
105#	2327	(N,0.29)
106#	854	(N,0.29)
107#	4977	(N,0.29)
108#	1234	(N,0.29)
109#	4913	(N,0.29)
110#	1421	(N,0.29)
111#	802	(N,0.28)
112#	3136	(N,0.28)
113#	4863	(N,0.27)
114#	185	(N,0.27)

Tabla B.13: Evaluación de $\langle examin, VH, \_, \_\rangle$ con SRI'_{2t} (Continuación III).

	RESULTADOS DE LA CONSULTA	
	Parametros utilizados	
Jerarquia = (3,3)	*Base de Datos* = "trec"(5000 Docs.)	*orness* = _
	Numero de documentos recuperados = 143	
Rank	**ID Doc**	**RSV**
115#	1757	(N,0.27)
116#	4103	(N,0.27)
117#	3395	(N,0.26)
118#	993	(N,0.26)
119#	3777	(N,0.26)
120#	4464	(N,0.25)
121#	2454	(N,0.24)
122#	2074	(N,0.24)
123#	3193	(N,0.24)
124#	2364	(N,0.24)
125#	235	(N,0.24)
126#	1499	(N,0.23)
127#	2760	(N,0.23)
128#	3984	(N,0.23)
129#	2423	(N,0.23)
130#	1816	(N,0.22)
131#	129	(N,0.22)
132#	3658	(N,0.22)
133#	971	(N,0.21)
134#	3693	(N,0.21)
135#	640	(N,0.20)
136#	630	(N,0.20)
137#	4038	(N,0.20)
138#	4777	(N,0.20)
139#	3301	(N,0.20)
140#	1374	(N,0.18)
141#	842	(N,0.18)
142#	4985	(N,0.17)
143#	3766	(N,0.15)

Tabla B.14: Evaluación de $\langle examin, VH, \_, \_ \rangle$ con SRI'_{2t} (Continuación IV).

	RESULTADOS DE LA CONSULTA	
	Parametros utilizados	
Jerarquia = (3,3)	*Base de Datos* = "trec"(5000 Docs.)	*orness* = _
	Numero de documentos recuperados = 31	
Rank	**ID Doc**	**RSV**
1#	2120	(VL,-0.18)
2#	4148	(EL,0.17)
3#	2023	(EL,0.05)
4#	339	(EL,-0.04)
5#	4745	(EL,-0.09)
6#	305	(EL,-0.10)
7#	432	(EL,-0.18)
8#	1312	(EL,-0.21)
9#	244	(EL,-0.22)
10#	3325	(EL,-0.22)
11#	2525	(EL,-0.24)
12#	1724	(EL,-0.25)
13#	989	(EL,-0.28)
14#	844	(EL,-0.28)
15#	3142	(EL,-0.32)
16#	2078	(EL,-0.34)
17#	1421	(EL,-0.39)
18#	185	(EL,-0.39)
19#	4288	(EL,-0.39)
20#	1621	(EL,-0.40)
21#	3131	(EL,-0.42)
22#	3312	(EL,-0.44)
23#	3308	(EL,-0.44)
24#	4234	(EL,-0.44)
25#	1784	(EL,-0.47)
26#	4518	(EL,-0.48)
27#	1384	(N,0.49)
28#	4325	(N,0.44)
29#	4542	(N,0.36)
30#	2031	(N,0.36)
31#	3766	(N,0.35)

Tabla B.15: Evaluación de $\langle jordan, M, \_, \_\rangle$ con SRI'_{2t}.

	RESULTADOS DE LA CONSULTA	
	Parametros utilizados	
Jerarquia = (3,3)	*Base de Datos* = "trec"(5000 Docs.)	*orness* = 1.00
	Numero de documentos recuperados = 2	
Rank	**ID Doc**	**RSV**
1#	2423	(M,-0.22)
2#	185	(M,-0.25)

Tabla B.16: Evaluación de $\langle bay, N, \_, \_\rangle AND \langle clamp, L, \_, \_\rangle$ con SRI'_{2t} con $orness = 1.0$.

	RESULTADOS DE LA CONSULTA	
	Parametros utilizados	
Jerarquia = (3,3)	*Base de Datos* = "trec"(5000 Docs.)	*orness* = 0.50
	Numero de documentos recuperados = 58	
Rank	**ID Doc**	**RSV**
1#	2423	(VH,-0.48)
2#	185	(H,0.46)
3#	1980	(M,-0.35)
4#	1816	(M,-0.37)
5#	4984	(M,-0.43)
6#	4097	(M,-0.44)
7#	2621	(M,-0.47)
8#	4459	(M,-0.49)
9#	4157	(L,0.50)
10#	4782	(L,0.44)
11#	4133	(L,0.33)
12#	3030	(L,0.17)
13#	4220	(L,0.16)
14#	4955	(VL,-0.09)
15#	297	(VL,-0.09)
16#	1196	(VL,-0.10)
17#	2843	(VL,-0.10)
18#	4493	(VL,-0.10)
19#	1225	(VL,-0.11)
20#	4742	(VL,-0.11)
21#	3370	(VL,-0.13)
22#	1764	(VL,-0.13)
23#	3331	(VL,-0.13)
24#	4606	(VL,-0.13)
25#	2929	(VL,-0.13)
26#	1913	(VL,-0.13)
27#	3824	(VL,-0.13)
28#	284	(VL,-0.14)
29#	1922	(VL,-0.14)
30#	3467	(VL,-0.14)

Tabla B.17: Evaluación de $\langle bay, N, \_, \_\rangle AND \langle clamp, L, \_, \_\rangle$ con SRI'_{2t} y $orness = 0.5$.

B.3. Más Ejemplos de Rendimiento de con SRI_{un}

	RESULTADOS DE LA CONSULTA	
	Parametros utilizados	
Jerarquia = (3,3)	*Base de Datos* = "trec"(5000 Docs.)	*orness* = 0.50
	Numero de documentos recuperados = 58	
Rank	**ID Doc**	**RSV**
31#	4355	(VL,-0.14)
32#	92	(VL,-0.14)
33#	4528	(VL,-0.14)
34#	3378	(VL,-0.15)
35#	2618	(VL,-0.15)
36#	4569	(VL,-0.15)
37#	4964	(VL,-0.15)
38#	2385	(VL,-0.15)
39#	1749	(VL,-0.16)
40#	3861	(VL,-0.17)
41#	601	(VL,-0.18)
42#	3374	(VL,-0.19)
43#	3517	(VL,-0.19)
44#	2633	(VL,-0.21)
45#	4703	(VL,-0.21)
46#	187	(VL,-0.24)
47#	4720	(VL,-0.25)
48#	4733	(VL,-0.29)
49#	4558	(VL,-0.30)
50#	4265	(VL,-0.31)
51#	4057	(VL,-0.31)
52#	1337	(VL,-0.31)
53#	2564	(VL,-0.42)
54#	3886	(VL,-0.43)
55#	4391	(VL,-0.49)
56#	2973	(VL,-0.50)
57#	4724	(EL,0.46)
58#	4669	(EL,0.43)

Tabla B.18: Evaluación de $\langle bay, N, \_, \_\rangle AND \langle clamp, L, \_, \_\rangle$ con SRI'_{2t} y $orness = 0.5$ (Continuación).

	RESULTADOS DE LA CONSULTA	
	Parametros utilizados	
Jerarquia = (3,3)	*Base de Datos* = "trec"(5000 Docs.)	*orness* = 0.00
	Numero de documentos recuperados = 4	
Rank	**ID Doc**	**RSV**
1#	185	(EL,-0.39)
2#	1922	(N,0.31)
3#	2423	(N,0.23)
4#	1816	(N,0.22)

Tabla B.19: Evaluación de $(\langle bay, N, T, VL\rangle OR\langle clamp, L, T, H\rangle) AND(\langle examin, VH, T, T\rangle$ $OR\langle jordan, M, T, T\rangle)$.

	RESULTADOS DE LA CONSULTA	
	Parametros utilizados	
Jerarquia = (1,3)	*Base de Datos* = "trec"(5000 Docs.)	*orness* = _
	Numero de documentos recuperados = 143	
Rank	**ID Doc**	**RSV**
1#	2125	(N,0.32)
2#	2782	(N,0.30)
3#	3096	(N,0.28)
4#	2132	(N,0.24)
5#	788	(N,0.23)
6#	3364	(N,0.22)
7#	1242	(N,0.21)
8#	4776	(N,0.21)
9#	2497	(N,0.20)
10#	1494	(N,0.20)
11#	486	(N,0.19)
12#	2380	(N,0.19)
13#	4940	(N,0.18)
14#	1627	(N,0.18)
15#	4865	(N,0.18)
16#	1426	(N,0.17)
17#	4216	(N,0.17)
18#	2458	(N,0.16)
19#	3357	(N,0.16)
20#	251	(N,0.16)
21#	1063	(N,0.15)
22#	1246	(N,0.15)
23#	3266	(N,0.15)
24#	3027	(N,0.14)
25#	3576	(N,0.14)
26#	3453	(N,0.14)
27#	2521	(N,0.14)
28#	4399	(N,0.14)
29#	2509	(N,0.14)
30#	871	(N,0.14)

Tabla B.20: Evaluación de $\langle examin, VH, \_, \_ \rangle$ con SRI_{un}.

	RESULTADOS DE LA CONSULTA	
	Parametros utilizados	
Jerarquia = (1,3)	*Base de Datos* = "trec"(5000 Docs.)	*orness* = _
	Numero de documentos recuperados = 143	
Rank	**ID Doc**	**RSV**
31#	2466	(N,0.14)
32#	383	(N,0.13)
33#	4204	(N,0.13)
34#	2866	(N,0.13)
35#	150	(N,0.13)
36#	4885	(N,0.12)
37#	2694	(N,0.12)
38#	756	(N,0.12)
39#	4706	(N,0.12)
40#	2291	(N,0.12)
41#	241	(N,0.12)
42#	2609	(N,0.12)
43#	958	(N,0.12)
44#	1209	(N,0.12)
45#	4131	(N,0.11)
46#	3843	(N,0.11)
47#	4364	(N,0.11)
48#	1415	(N,0.11)
49#	4621	(N,0.11)
50#	1136	(N,0.11)
51#	381	(N,0.11)
52#	4727	(N,0.11)
53#	2161	(N,0.11)
54#	1060	(N,0.11)
55#	463	(N,0.10)
56#	4968	(N,0.10)
57#	3322	(N,0.10)
58#	1124	(N,0.10)
59#	4301	(N,0.10)
60#	1890	(N,0.10)

Tabla B.21: Evaluación de $\langle examin, VH, \_, \_ \rangle$ con SRI_{un} (Continuación I).

	RESULTADOS DE LA CONSULTA	
	Parametros utilizados	
Jerarquia = (1,3)	*Base de Datos* = "trec"(5000 Docs.)	*orness* = _
	Numero de documentos recuperados = 143	
Rank	**ID Doc**	**RSV**
61#	3823	(N,0.10)
62#	2422	(N,0.09)
63#	3397	(N,0.09)
64#	1238	(N,0.09)
65#	1458	(N,0.09)
66#	2271	(N,0.09)
67#	2342	(N,0.09)
68#	4986	(N,0.09)
69#	3380	(N,0.09)
70#	2840	(N,0.09)
71#	1082	(N,0.09)
72#	725	(N,0.09)
73#	422	(N,0.09)
74#	3130	(N,0.09)
75#	3541	(N,0.09)
76#	1405	(N,0.09)
77#	1108	(N,0.09)
78#	3932	(N,0.09)
79#	2044	(N,0.08)
80#	1252	(N,0.08)
81#	4553	(N,0.08)
82#	3485	(N,0.08)
83#	1794	(N,0.08)
84#	2395	(N,0.08)
85#	3906	(N,0.08)
86#	3448	(N,0.08)
87#	3424	(N,0.08)
88#	3549	(N,0.08)
89#	1026	(N,0.08)
90#	3665	(N,0.08)

Tabla B.22: Evaluación de $\langle examin, VH, \_, \_\rangle$ con SRI_{un} (Continuación II).

	RESULTADOS DE LA CONSULTA	
	Parametros utilizados	
Jerarquia = (1,3)	*Base de Datos* = "trec"(5000 Docs.)	*orness* = _
	Numero de documentos recuperados = 143	
Rank	**ID Doc**	**RSV**
91#	4299	(N,0.08)
92#	1908	(N,0.08)
93#	2269	(N,0.08)
94#	273	(N,0.08)
95#	2445	(N,0.08)
96#	1922	(N,0.08)
97#	2066	(N,0.08)
98#	3456	(N,0.08)
99#	1604	(N,0.08)
100#	1689	(N,0.08)
101#	2374	(N,0.08)
102#	2467	(N,0.07)
103#	4172	(N,0.07)
104#	3439	(N,0.07)
105#	2327	(N,0.07)
106#	854	(N,0.07)
107#	4977	(N,0.07)
108#	1234	(N,0.07)
109#	4913	(N,0.07)
110#	1421	(N,0.07)
111#	802	(N,0.07)
112#	3136	(N,0.07)
113#	4863	(N,0.07)
114#	185	(N,0.07)
115#	1757	(N,0.07)
116#	4103	(N,0.07)
117#	3395	(N,0.07)
118#	993	(N,0.07)
119#	3777	(N,0.07)
120#	4464	(N,0.06)

Tabla B.23: Evaluación de $\langle examin, VH, \_, \_\rangle$ con SRI_{un} (Continuación III).

	RESULTADOS DE LA CONSULTA	
	Parametros utilizados	
Jerarquia = (1,3)	*Base de Datos* = "trec"(5000 Docs.)	*orness* = _
	Numero de documentos recuperados = 143	
Rank	**ID Doc**	**RSV**
121#	2454	(N,0.06)
122#	2074	(N,0.06)
123#	3193	(N,0.06)
124#	2364	(N,0.06)
125#	235	(N,0.06)
126#	1499	(N,0.06)
127#	2760	(N,0.06)
128#	3984	(N,0.06)
129#	2423	(N,0.06)
130#	1816	(N,0.06)
131#	129	(N,0.05)
132#	3658	(N,0.05)
133#	971	(N,0.05)
134#	3693	(N,0.05)
135#	640	(N,0.05)
136#	630	(N,0.05)
137#	4038	(N,0.05)
138#	4777	(N,0.05)
139#	3301	(N,0.05)
140#	1374	(N,0.05)
141#	842	(N,0.05)
142#	4985	(N,0.04)
143#	3766	(N,0.04)

Tabla B.24: Evaluación de $\langle examin, VH, \_, \_ \rangle$ con SRI_{un} (Continuación IV).

	RESULTADOS DE LA CONSULTA	
	Parametros utilizados	
Jerarquia = (1,3)	*Base de Datos* = "trec"(5000 Docs.)	*orness* = _
	Numero de documentos recuperados = 31	
Rank	**ID Doc**	**RSV**
1#	2120	(N,0.46)
2#	4148	(N,0.29)
3#	2023	(N,0.26)
4#	339	(N,0.24)
5#	4745	(N,0.23)
6#	305	(N,0.22)
7#	432	(N,0.21)
8#	1312	(N,0.20)
9#	244	(N,0.20)
10#	3325	(N,0.20)
11#	2525	(N,0.19)
12#	1724	(N,0.19)
13#	989	(N,0.18)
14#	844	(N,0.18)
15#	3142	(N,0.17)
16#	2078	(N,0.17)
17#	1421	(N,0.15)
18#	185	(N,0.15)
19#	4288	(N,0.15)
20#	1621	(N,0.15)
21#	3131	(N,0.14)
22#	3312	(N,0.14)
23#	3308	(N,0.14)
24#	4234	(N,0.14)
25#	1784	(N,0.13)
26#	4518	(N,0.13)
27#	1384	(N,0.12)
28#	4325	(N,0.11)
29#	4542	(N,0.09)
30#	2031	(N,0.09)
31#	3766	(N,0.09)

Tabla B.25: Evaluación de $\langle jordan, M, \_, \_ \rangle$ con SRI_{un}.

	RESULTADOS DE LA CONSULTA	
	Parametros utilizados	
Jerarquia = (2,3)	*Base de Datos* = "trec"(5000 Docs.)	*orness* = 1.00
	Numero de documentos recuperados = 2	
Rank	**ID Doc**	**RSV**
1#	2423	(M,-0.11)
2#	185	(M,-0.12)

Tabla B.26: Evaluación de $\langle bay, N, \_, \_\rangle AND \langle clamp, L, \_, \_\rangle$ con SRI_{un}.

	RESULTADOS DE LA CONSULTA	
	Parametros utilizados	
Jerarquia = (2,3)	*Base de Datos* = "trec"(5000 Docs.)	*orness* = 0.50
	Numero de documentos recuperados = 58	
Rank	**ID Doc**	**RSV**
1#	2423	(H,0.34)
2#	185	(H,0.25)
3#	1980	(M,-0.26)
4#	1816	(M,-0.28)
5#	4984	(M,-0.32)
6#	4097	(M,-0.33)
7#	2621	(M,-0.35)
8#	4459	(M,-0.37)
9#	4157	(M,-0.38)
10#	4782	(M,-0.42)
11#	4133	(L,0.49)
12#	3030	(L,0.37)
13#	4220	(L,0.37)
14#	4955	(L,-0.04)
15#	297	(L,-0.05)
16#	1196	(L,-0.05)
17#	2843	(L,-0.05)
18#	4493	(L,-0.05)
19#	1225	(L,-0.06)
20#	4742	(L,-0.06)
21#	3370	(L,-0.06)
22#	1764	(L,-0.06)
23#	3331	(L,-0.06)
24#	4606	(L,-0.06)
25#	2929	(L,-0.06)
26#	1913	(L,-0.07)
27#	3824	(L,-0.07)
28#	284	(L,-0.07)
29#	1922	(L,-0.07)
30#	3467	(L,-0.07)

Tabla B.27: Evaluación de $\langle bay, N, \_, \_\rangle AND \langle clamp, L, \_, \_\rangle$ con SRI_{un} y $orness = 0.5$.

	RESULTADOS DE LA CONSULTA	
	Parametros utilizados	
Jerarquia = (2,3)	*Base de Datos* = "trec"(5000 Docs.)	*orness* = 0.50
	Numero de documentos recuperados = 58	
Rank	**ID Doc**	**RSV**
31#	4355	(L,-0.07)
32#	92	(L,-0.07)
33#	4528	(L,-0.07)
34#	3378	(L,-0.07)
35#	2618	(L,-0.08)
36#	4569	(L,-0.08)
37#	4964	(L,-0.08)
38#	2385	(L,-0.08)
39#	1749	(L,-0.08)
40#	3861	(L,-0.08)
41#	601	(L,-0.09)
42#	3374	(L,-0.09)
43#	3517	(L,-0.09)
44#	2633	(L,-0.10)
45#	4703	(L,-0.11)
46#	187	(L,-0.12)
47#	4720	(L,-0.13)
48#	4733	(L,-0.14)
49#	4558	(L,-0.15)
50#	4265	(L,-0.15)
51#	4057	(L,-0.15)
52#	1337	(L,-0.16)
53#	2564	(L,-0.21)
54#	3886	(L,-0.21)
55#	4391	(L,-0.24)
56#	2973	(L,-0.25)
57#	4724	(L,-0.27)
58#	4669	(L,-0.28)

Tabla B.28: Evaluación de $\langle bay, N, \_, \_\rangle AND \langle clamp, L, \_, \_\rangle$ con SRI_{un} y $orness = 0.5$ (Continuación).

	RESULTADOS DE LA CONSULTA	
	Parametros utilizados	
Jerarquia = (2,3)	*Base de Datos* = "trec"(5000 Docs.)	*orness* = 0.00
	Numero de documentos recuperados = 4	
Rank	**ID Doc**	**RSV**
1#	185	(N,0.30)
2#	1922	(N,0.15)
3#	2423	(N,0.11)
4#	1816	(N,0.11)

Tabla B.29: Evaluación de $(\langle bay, N, T, M\rangle OR\langle clamp, L, T, M\rangle) AND(\langle examin, VH, T, M\rangle$ $OR\langle jordan, M, T, M\rangle)$ con SRI_{un}.

Bibliografía

[1] *Some examples of stoplist [en línea]. http://terral.lsi.uned.es/ircourse/examples/stoplist.html [consulta: 16 de enero de 2006].*

[2] B. Arfi, *Fuzzy decision making in politics: A linguistic fuzzy-set approach (*LFSA*)*, Political Analysis **13** (2005), no. 1, 23–56.

[3] R. Baeza-Yates, *Information retrieval in the web: Beyond current search engines*, International Journal of Approximate Reasoning **34** (2003), no. 2–3, 97–104.

[4] R. Baeza-Yates and B. Ribeiro-Neto, *Modern information retrieval*, Addison-Wesley, 1999.

[5] N.J. Belkin and W.B Croft, *Information filtering and information retrieval: Two sides of the same coin?*, Communications of the ACM **35** (1992), no. 12, 29–38.

[6] L. Bolc, A. Kowalski, and M. Kozlowska, *A natural language information retrieval system with extensions towards fuzzy reasoning*, Internationa Journal of Man-Machine Studies **23** (1985), no. 4, 335–367.

[7] P.P Bonissone and K.S. Decker, *Uncertainty in artificial intelligence*, ch. Selecting Uncertainty Calculi and Granularity: An Experiment in Trading-off Precision and Complexity, pp. 217–247, L.H. Kanal and J.F. Lemmer, Eds. (North–Holland), 1986.

[8] A. Bookstein, *Fuzzy request: An approach to weighted boolean searches*, Journal of the American Society for Information Science and Technology (1980), no. 31, 240–247.

[9] ———, *Outline of a general probabilistic retrieval model*, Journal of Documentation **39** (1983), no. 2, 63–72.

[10] G. Bordogna and G. Carrara, P. Pasi, *Query term weights as constraints in fuzzy information retrieval*, Information Processing & Management **27** (1991), no. 1, 15–26.

[11] G. Bordogna, P. Carrara, and G. Pasi, *Extending boolean information retrieval: A fuzzy model based on linguistic variables*, Proceedings of the First IEEE International Conference on Fuzzy Systems. San Diego, California, 1992, pp. 769–776.

[12] ———, *Fuzzy sets and possibility theory in database management systems*, ch. Fuzzy Approaches to Extend Boolean Information Retrieval, pp. 231–274, Bosc, P. and Kacpryk, J., Eds. (Springer Verlag), 1995.

[13] G. Bordogna and G. Pasi, *A fuzzy linguistic approach generalizing boolean information retrieval: A model and its evaluation*, Journal of the American Society for Information Science **44** (1993), no. 2, 70–82.

[14] ———, *Linguistic aggregation operators of selecction criteria in fuzzy information retrieval*, International Journal of Intelligent Systems **10** (1995), 233–248.

[15] ———, *An ordinal information retrieval model*, International Journal of Uncertainty, Fuzziness and Knowledge-Based Systems **9** (2001), 63–76.

[16] G. Bordogna, G. Pasi, *The ordered weighted averaging operators: Theory and applications*, ch. Application of the OWA operators to soften information re-

trieval systems, pp. 275–294, Yager, R.R. and Kacprzyk, J., Eds. (Kluwer Academic Publishers), 1997.

[17] ———, *Soft computing in information retrieval*, ch. Application of Fuzzy Sets Theory to Extend Boolean Information Retrieval, pp. 21–47, Crestani, F. and Pasi, G., Eds. (Physica Verlag), 2000.

[18] C. Buckley, *Implementation of the smart information retrieval system*, Technical Report TR85–686, Universidad de Cornell, 1985.

[19] D. Buell and D.H. Kraft, *A model for a weighted retrieval system*, Journal of the American Society for Information Science **32** (1981), 211–216.

[20] ———, *Threshold values and boolean retrieval systems*, Information Processing & Management **17** (1981), 127–136.

[21] C.S. Cater and D.H. Kraft, *A generalization and clarification of the* W*aller-*K*raft wish-list*, Information Processing & Management **25** (1989), 15–25.

[22] H. Chen, *Preface to the special issue: "web retrieval and mining"*, Decision Support Systems **35** (2003), 1–5.

[23] H. Chen, G. Shankaranarayanan, L. She, and A. Iyer, *A machine learning approach to inductive query by example: An experiment using relevance feedback,* ID3*, genetic algorithms, and simulated annealing*, Journal of the American Society for Information Science **49** (1998), no. 8, 693–705.

[24] C.W. Cleverdon, *On the inverse relationship of recall and precision*, Journal of Documentation **28** (1972), 195–201.

[25] C.W. Cleverdon and E.M. Keen, *Factors determining the performance of indexing systems*, Technical report, College of Aeronautics, Cranfield, UK, 1966.

[26] O. Cordón, F. Herrera, and I. Zwir, *Linguistic modeling by hierarchical systems of linguistic rules*, IEEE Transactions on Fuzzy Systems **10** (2002), no. 1, 2–20.

[27] O. Cordón and E. Herrera-Viedma, *Preface to the special issue on soft computing applications to intelligent information retrieval on the internet*, International Journal of Approximate Reasoning **34** (2003), no. 2–3, 89–95.

[28] F. Crestani and G. Pasi, *Handling vagueness, subjectivity, and imprecision in information access: An introducction to the special issue*, Information Processing & Management **39** (2003), no. 2, 161–165.

[29] W.B. Croft and D.J. Harper, *Using probabilistic models of document retrieval without relevance information*, Journal of Documentation **35** (1979), no. 4, 285–295.

[30] R. Degani and G. Bortolan, *The problem of linguistic approximation in clinical decision making*, International Journal of Approximate Reasoning **2** (1988), 143–162.

[31] M. Delgado, F. Herrera, E. Herrera-Viedma, and L. Martínez, *Combining numerical and linguistic information in group decision making*, Information Sciences **107** (1998), 177–194.

[32] M. Delgado, J.L. Verdegay, and M.A. Vila, *On aggregation operations of linguistic labels*, International Journal of Intelligent Systems **8** (1993), 351–370.

[33] B. Dervin and M.Ñilan, *Information needs and uses*, Annual Review of Information Science and Technology **21** (1986), 3–33.

[34] D. Ellis, *The physical and cognitve paradigms in information retrieval research*, Journal of Documentation **48** (1992), no. 1, 45–64.

[35] W. Fan, M.D. Gordon, and P. Pathak, *Effective profiling of consumer information retrieval needs: A unified framework and empirical comparision*, Decision Support Systems (2005), por aparecer.

[36] J. Fodor and M. Roubens, *Fuzzy preference modelling and multicriteria decision support*, Kluwer Academic Publishers, 1994.

[37] C. Fox, *Information retrieval. data structures & algorithms*, ch. Lexical analysis and stoplist, pp. 102–130, Frakes, W. and Baeza-Yates, R., Eds. (Prentice-Hall), New Jersey, 1992.

[38] W. Frakes, *Information retrieval. data structures & algorithms*, ch. Stemming Algorithms, pp. 131–160, Frakes, W. and Baeza-Yates, R., Eds. (Prentice-Hall), New Jersey, 1992.

[39] N. Fuhr and C. Buckley, *A probabilistic learning approach for document indexing*, ACM Transactions on Information Systems **9** (1991), no. 3, 223–248.

[40] G.W. Furnas, T.K. Landauer, L.M. Gomez, and S.T. Dumais, *The vocabulary problem in human-system communication*, Communications of ACM **30** (1987), no. 11, 947–971.

[41] U. Hanani, B. Shapira, and P. Shoval, *Information filtering: Overview of issues, research and systems*, User Modeling and User-Adapted Interaction **11** (2001), no. 3, 203–259.

[42] F. Herrera and E. Herrera-Viedma, *Aggregation operators for linguistic weighted information*, IEEE Transactions on Systems, Man, and Cybernetics, Part A: Systems and Humans **27** (1997), 646–656.

[43] F. Herrera, E. Herrera-Viedma, and L. Martínez, *A fusion approach for managing multigranularity linguistic terms sets in decision making*, Fuzzy Sets and Systems **114** (2000), 43–58.

[44] F. Herrera, E. Herrera-Viedma, and J.L. Verdegay, *Direct approach processes in group decision making using linguistic owa operators*, Fuzzy Sets and Systems **79** (1996), 175–190.

[45] ———, *Direct approach processes in group decision making using linguistic owa operators*, Fuzzy Sets and Systems **79** (1996), 175–190.

[46] F. Herrera and L. Martínez, *A 2-tuple fuzzy linguistic representation model for computing with words*, IEEE Transactions on Fuzzy Systems **8** (2000), no. 6, 746–752.

[47] F. Herrera and L. Martínez, *A 2-tuple fuzzy linguistic representation model for computing with words*, IEEE Transactions on Fuzzy Systems **8** (2000), no. 6, 746–752.

[48] F. Herrera and L. Martínez, *The 2-tuple linguistic computational model. advantages of its linguistic description, accuracy and consistency*, International Journal of Uncertainty , Fuzziness and Knowledge-Based Systems **9** (2001), 33–48.

[49] F. Herrera and L. Matínez, *A model based on linguistic 2-tuples for dealing with multigranularity hierarchical linguistic contexts in multiexpert decision-making*, IEEE Transactions on Systems, Man and Cybernetics. Part B: Cybernetics **31** (2001), no. 2, 227–234.

[50] E. Herrera-Viedma, *An information retrieval system with ordinal linguistic weighted queries based on two weighting elements*, International Journal of

Uncertainty, Fuzziness and Knowledge-Based Systems **9** (2001), 77–88.

[51] ———, *Modelling the retrieval process for an information retrieval system using an ordinal fuzzy linguistic approach*, Journal of the American Society for Information Science and Technology **52** (2001), no. 6, 460–475.

[52] E. Herrera-Viedma, O. Cordón, J.C. Herrera, and M. Luque, *An irs based on multi-granular linguistic information*, Proceedings of the 7th International Conference of the International Society for Knowledge Organization (ISKO 2002), 2002, pp. 372–378.

[53] E. Herrera-Viedma, O. Cordón, M. Luque, A.G. López, and A.M. Muñoz, *A model of fuzzy linguistic irs based on multi-granular linguistic information*, International Journal of Approximate Reasoning **34** (2003), 221–239.

[54] E. Herrera-Viedma, L. Martínez, F. Mata, and F. Chiclana, *A consensus support system model for group decision-making problems with multi-granular linguistic preference relations*, IEEE Transaction on Fuzzy Systems (2005), por aparecer.

[55] E. Herrera-Viedma and E. Peis, *Evaluating the informative quality of documents in* SGML*-format using fuzzy linguistic techniques based on computing with words*, Information Processing & Management **39** (2003), no. 2, 195–213.

[56] P. Ingwersen and P. Willet, *An introduction to algorithmic and cognitive approaches for information retrieval*, Libri **45** (1995), no. 3–4, 169–177.

[57] K.S. Jones, *Search term relevance weighting given very little relevance information*, Journal of Documentation **35** (1979), no. 1, 30–48.

[58] G.J. Klir and B. Yuan, *Fuzzy sets and fuzzy logic: Theory and applications*, Prentice Hall, 1995.

[59] M. Kobayashi and K. Takeda, *Information retrieval on the web*, ACM Computing Surveys **32** (2000), no. 2, 144–173.

[60] R. Korfhage, *Information storage and retrieval*, Wiley, New York, 1997.

[61] D.H. Kraft, G. Bordogna, and G. Pasi, *An extended fuzzy linguistic approach to generalize boolean information retrieval*, Information Sciences **2** (1994), 119–134.

[62] D.H. Kraft and D.A. Buell, *Fuzzy sets and generalized boolean retrieval systems*, International Journal of Man-Machine Studies **19** (1983), 45–56.

[63] F.W. Lancaster, *Information retrieval systems - characteristics, testing and evaluation*, ch. Criteria by Which Information Retrieval Sytstems May Be Evaluated, Nueva York: Willey, 1979.

[64] S. Lawrence and C.L. Giles, *Searching the world wide web*, Science **280** (1998), no. 5360, 98–100.

[65] ———, *Searching the web: General and scientific information access*, IEEE Communications Magazine **37** (1999), no. 1, 116–122.

[66] G. Marchioni, *Information seeking in electronic environments*, Cambridge University Press, 1995.

[67] M.E. Maron and J.L. Kuhns, *On relevance, probabilistic indexing and information retrieval*, Journal of the Association for Computer Machinery **7** (1960), 216–244.

[68] G.A. Miller, *The magical number seven or minus two: Some limits on our capacity of processing information*, Psychological Rev. **63** (1956), 81–97.

[69] T. Mitchell, *Machine learning*, Mc-Graw Hill, 1997.

[70] S. Miyamoto, *Fuzzy sets in information retrieval and cluster analysis*, Kluwer Academic Publishers, 1990.

[71] D.W. Oard and G. Marchionini, *A conceptual framework for text filtering*, University of Maryland, College Park, CS-TR-3643, 1996.

[72] M.D. Olvera, *Evaluación de la recuperación de información en internet: Un modelo experimental*, Tesis doctoral, Universidad de Granada, 1999.

[73] M.L. Pao, *Automatic text analysis based on transition phenomena of word occurrences*, Journal of American Society for Information Science (JASIS) **1978** (1978), 121–124.

[74] G. Pasi, J.M. Benítez, O. Cordón, F. Hoffman, and R. Roy, *Advances in soft computing. engineering design and manufacturing*, ch. Intelligent Information Retrieval: Some Research Trends, pp. 157–171, Pasi, G. and Benítez, J.M. and Cordón, O. and Hoffman, F. and Roy, R., Eds. (Springer), 2003.

[75] P. Perny and J.D. Zucker, *Preference-based search and machine learning for collaborative filtering: the film conseil movie recommender system*, Information - Interaction - Intelligence **1** (2001), no. 1, 13.

[76] M.F. Porter, *An algorithm for suffix stripping*, Program **14** (1980), no. 3, 130–137.

[77] T. Radecki, *Fuzzy set theoretical approach to document retrieval information*, Information Processing & Management (1979), no. 15, 247–260.

[78] S.E. Robertson, *The probability ranking principle in* IR, Journal of Documentation **33** (1977), no. 4, 294–304.

[79] S.E. Robertson and K.S. Jones, *Relevance weighting of search terms*, Journal of the American Society for Information Science **27** (1976), no. 3, 129–146.

[80] G. Salton, *Automatic text processing: The transformation, analysis and retrieval of information by computer*, Addison-Wesley, 1989.

[81] G. Salton and C. Buckley, *Term-weighting approaches in automatic text retrieval*, Information Processing & Management **24** (1988), 513–523.

[82] G. Salton and M.E. Lesk, *Computer evaluation of indexing and text precising*, Journal of the Association of Computing Machinery **15** (1968), 8–36.

[83] G. Salton and M.J. McGill, *An introduction to modern information retrieval*, McGraw-Hill, 1983.

[84] E. Sanchez, *Importance in knowledge-based systems*, Information Systems **14** (1989), no. 6, 455–464.

[85] C.K. Schultz, H.P. L*uhn: Pioneer of information science*, Spartan Books, New York, NY, 1968.

[86] L.C. Smith, *Artificial intelligence and information retrieval*, Annual Review of Information Science and Technology **22** (1987), 41–77.

[87] V. Torra, *Negation functions based semantics for ordered linguistic labels*, International Journal of Intelligent Systems **11** (1996), 975–988.

[88] ———, *Aggregation of linguistic labels when semantics is based on antonyms*, International Journal of Intelligent Systems **16** (2001), 513–524.

[89] H.R. Turtle, *Inference networks for document retrieval*, Tesis doctoral, Universidad de Massachusetts, 1990.

[90] C.J. van Rijsbergen, *Information retrieval*, Butterworth, 1979.

[91] W.G. Waller and D.H. Kraft, *A mathematical model of a weighted boolean retrieval system*, Information Processing & Management **15** (1979), 235–245.

[92] R.R. Yager, *A note on weighted queries in information retrieval systems*, Journal of the American Society of Information Sciences **38** (1987), 23–24.

[93] ———, *On ordered weighted averaging aggregation operators in multicriteria decision making*, IEEE Transactions on Systems, Man, and Cybernetics **18** (1988), 183–190.

[94] ———, *An approach to ordinal decision making*, International Journal of Approximate Reasoning **12** (1995), 237–261.

[95] ———, *A hierarchical document retrieval languaje*, Information Retrieval **3** (2000), 357–377.

[96] L. A. Zadeh, *Web intelligence and fuzzy logic-the concept of web* IIQ *(*WIQ*) [en línea]*, 2003.

[97] L.A. Zadeh, *The concept of a linguistic variable and its applications to approximate reasoning.* P*art* I*, information sciences 8 (1975) 199-249,* P*art* II*, information sciences 8 (1975) 301-357,* P*art* III*, information sciences 9 (1975) 43-80.*

[98] ———, *Fuzzy sets*, Information and Control **8** (1965), no. 3, 338–353.

[99] ———, *A note on web intelligence, world knowledge and fuzzy logic*, Data and Knowledge Engineering **50** (2004), 291–304.

[100] L.A. Zadeh and J. Kacprzyk, *Fuzzy logic for the management of uncertainty*, John Wiley, New York, 1992.

[101] H.J. Zimmermann, *Fuzzy sets: Theory and its applications*, Kluwer Academic, 1996.

[102] G.K. Zipf and L. Thiele, *Human behavior and the principle of least effort*, Addison Wesley, Cambridge, 1949.