

**DIVISIÓN DE ESTUDIOS DE POSGRADO E
INVESTIGACIÓN**

TESIS

**“ANÁLISIS DE MODELOS DE CARACTERIZACIÓN DE ARGUMENTOS DE UN
TOMADOR DE DECISIONES PARA SISTEMAS DE RECOMENDACIÓN”**

QUE PARA OBTENER EL GRADO DE:
MAESTRO EN CIENCIAS EN COMPUTACIÓN

PRESENTA:

**I.S.C. CÉSAR ALEJANDRO GUERRERO NAVA
G09070933**

DIRECTOR DE TESIS:
DR. NELSON RANGEL VALDEZ

CO-DIRECTORA DE TESIS:
DRA. LAURA CRUZ REYES

COMITÉ TUTORIAL:
**DRA. CLAUDIA GÓMEZ SANTILLÁN
DR. JUAN JAVIER GONZÁLEZ BARBOSA
DRA. GUADALUPE CASTILLA VALDEZ**



Cd. Madero, Tamps: a **28 de Septiembre de 2016**

OFICIO No.: US.170/16
AREA: DIVISION DE ESTUDIOS
DE POSGRADO E INVESTIGACIÓN
ASUNTO: AUTORIZACIÓN DE IMPRESIÓN DE TESIS

ING. CÉSAR ALEJANDRO GUERRERO NAVA
NO. DE CONTROL **G09070933**
PRESENTE

Me es grato comunicarle que después de la revisión realizada por el Jurado designado para su examen de grado de Maestría en Ciencias de la Computación, el cual está integrado por los siguientes catedráticos:

PRESIDENTE :	DRA. GUADALUPE CASTILLA VALDEZ
SECRETARIO :	DRA. CLAUDIA GUADALUPE GÓMEZ SANTILLÁN
VOCAL :	DR. NELSON RANGEL VALDEZ
SUPLENTE :	DR. JUAN JAVIER GONZÁLEZ BARBOSA
DIRECTOR DE TESIS:	DR. NELSON RANGEL VALDEZ
CO-DIRECTORA DE TESIS:	DRA. LAURA CRUZ REYES

Se acordó autorizar la impresión de su tesis titulada:

"ANÁLISIS DE MODELOS DE CARACTERIZACIÓN DE ARGUMENTOS DE UN TOMADOR DE DECISIONES PARA SISTEMAS DE RECOMENDACIÓN"

Es muy satisfactorio para la División de Estudios de Posgrado e Investigación compartir con Usted el logro de esta meta.

Espero que continúe con éxito su desarrollo profesional y dedique su experiencia e inteligencia en beneficio de México.

ATENTAMENTE
"POR MI PATRIA Y POR MI BENDICIÓN"

DRA. ADRIANA ISABEL REYES DE LA TORRE
JEFA DE LA DIVISIÓN DE ESTUDIOS
DE POSGRADO E INVESTIGACIÓN



S.E.P.
DIVISION DE ESTUDIOS
DE POSGRADO E
INVESTIGACION
I T C M

c.c.p.- Archivo
Minuta

ARTYCO



Ave. 1° de Mayo y Sor Juana I. de la Cruz Col. Los Manguos, C.P. 89440 Cd. Madero, Tam.
Tel. (833) 357 48 20. e-mail: itcm@itcm.edu.mx
www.itcmadu.mx



Declaración de Originalidad

Declaro y prometo que este documento de tesis es producto de mi trabajo original y que no infringe los derechos de terceros, tales como derecho de publicación, derechos de autor, patentes y similares.

Además, declaro que en las citas textuales que he incluido y en los resúmenes que he realizado de publicaciones ajenas, indico explícitamente los datos de los autores y las publicaciones.

Además, en caso de infracción a los derechos de terceros derivados de este documento de tesis, acepto la responsabilidad de la infracción y relevo de ésta a mi director y codirector de tesis, así como al Instituto Tecnológico de Cd. Madero y sus autoridades.

Noviembre de 2016, Cd. Madero, Tamaulipas.

I.S.C. César Alejandro Guerrero Nava

Agradecimientos

Reciban un sincero agradecimiento todos los integrantes del comité tutorial de esta tesis: Dr. Nelson Rangel Valdez, Dra. Laura Cruz Reyes, Dra. Claudia Guadalupe Gómez Santillán, Dr. Juan Javier González Barbosa y Dra. Guadalupe Castilla Valdez.

Agradezco sinceramente a mis directores de tesis: Dr. Nelson Rangel Valdez y a la Dra. Laura Cruz Reyes por haberme apoyado durante mi formación en la maestría, permitiéndome concluir con éxito.

También quiero agradecer a las instituciones que dieron todas las facilidades para que este trabajo se llevara a cabo: el Consejo Nacional de Ciencia y Tecnología (Conacyt) y al Instituto Tecnológico de Ciudad Madero, por haberme permitido continuar con un estudio de postgrado de alta calidad.

Gracias a todos mis compañeros y amigos de la maestría, por haberme dado la oportunidad de conocerlos y compartir nuevas y grandiosas experiencias.

Muchas gracias a mi familia, que estuvo apoyándome en todo tiempo y momento, tanto dentro como fuera del aula, animándome a no desistir, aunque las situaciones se tornaran complejas. Ellos serán testigos de cómo se cosecharán los frutos de su gran esfuerzo.

Y finalmente, pero no menos importante, gracias a Dios por permitirme concluir con mis estudios. Sino fuera porque Él es bueno y ha tenido misericordia, no habría llegado hasta aquí. Él ha sido fiel.

Resumen

La toma de decisiones es un proceso complejo, que si no es tomado con el debido cuidado, puede tener un efecto negativo en el desempeño. Aunque existen técnicas que ayudan a la toma de decisiones, estas siguen siendo insuficientes. Por ello, las ciencias de la computación tienen un impacto positivo al ayudar en la toma de decisiones. Muchas veces existe el obstáculo de que la persona o grupo de personas que toman las decisiones (mejor conocidas como DM, por sus siglas en inglés) necesitan estar presente para poder interactuar con Sistemas de Recomendación. En ocasiones, es posible que dicha interacción no sea posible, haciéndose una herramienta que asista al sistema de recomendación, en sustitución del DM. Por ello, el trabajo de tesis desarrollado, presenta una propuesta de arquitectura para implementar un Corpus de Diálogo Argumentativo. El objetivo de dicho corpus, es automatizar el proceso de interacción con un sistema de recomendación, de tal forma que emule a un Decisor (DM artificial), permitiendo validar si las recomendaciones establecidas para el problema en cuestión, son coherentes con las preferencias del decisor, llegando inclusive a cuestionar por medio de diferentes argumentos, para lograr resultados favorables. Los componentes que componen el proceso de caracterización son: el sistema experto (sistema de recomendación), la recomendación emitida por el sistema experto, los hechos (extraídos de la interacción), el corpus

de diálogo argumentativo (posee textos argumentativos que caracterizan a un decisor en particular), árboles de argumentos (textos argumentativos ordenados en una estructura de árbol), reglas de prolog, conclusiones, (junto con un método de selección) y actualización del conocimiento que surge en la interacción. Mientras mejor sea la caracterización del DM, mayor será la fidelidad entre el diálogo original y el emulado por el corpus. El trabajo aquí presentado, tiene como objetivo ayudar a la toma de decisiones, permitiendo la ausencia del decisor al elegir alternativas, dentro del ámbito del problema de cartera de proyectos públicos. A través de este trabajo de investigación, se logró automatizar la caracterización de un DM, permitiendo la interacción con un sistema de recomendación. Cabe mencionar que este procedimiento de interacción posee discrepancias con los diálogos originales, los cuales fueron realizados por el DM manualmente con el sistema de recomendación.

Tabla de Contenido

Declaración de Originalidad	ii
Agradecimientos	iii
Resumen	iv
Índice de Figuras	viii
Índice de Tablas	ix
Capítulo 1 Introducción	1
1.1. Antecedentes	1
1.2. Definición del problema	1
1.3. Justificación y beneficios.....	6
1.4. Planteamiento del Problema.....	8
1.5. Objetivo.....	8
1.5.1. Objetivo general	8
1.5.2. Objetivos específicos.....	8
1.6. Alcances y limitaciones	8
1.7. Organización del documento	9
Capítulo 2 Marco Teórico	10
2.1. Toma de decisiones	10
2.1.1. El problema de toma de decisión.....	10
2.1.2. Métodos de Análisis de Decisión multicriterio (MCDA).....	10
2.1.3. Modelo de Preferencias.....	11
2.2. Optimización Multiobjetivo	13
2.2.1. El problema de toma de decisión.....	13
2.2.2. Modelo matemático de un Problema Multiobjetivo.....	15
2.2.3. Estrategias de Optimización Multiobjetivo	16
2.3. Problema de Cartera de Proyectos Públicos.....	18
2.3.1. Proyecto.....	18
2.3.2. Características del Problema de Cartera de Proyectos Públicos.....	18
2.4. Teoría de Argumentación en la Toma de Decisiones.....	20
2.4.1. Sistema de recomendación	20
2.4.2. Teoría de la Argumentación	21

2.4.3. Juegos de Dialogo	22
2.5. Corpus de Diálogo Argumentativo	23
2.5.1. Definición	23
2.5.2. Modelo de Toulmin	23
2.5.3. Mapeo de Argumentos	23
Capítulo 3 Estado del Arte	25
3.1. Aprendizaje de la estructura de conversaciones orientadas a tareas desde un Corpus de diálogo de dominio específico	25
3.2. Generación de Argumentos con mayor credibilidad	25
3.3. Perspectiva Dialéctica en MCDA.....	26
3.4. Framework de Apoyo a la Toma de decisiones para el PCP	26
3.5. Análisis comparativo.....	26
Capítulo 4 Propuesta de Solución.....	28
4.1. Arquitectura General para la Caracterización de un DM.....	28
4.2. DM.....	29
4.2.1. Hechos.....	29
4.2.2. Corpus	29
4.2.3. Motor de inferencias.....	30
4.2.4. Conclusiones	31
4.2.5. Actualización.....	31
4.3. Sistema Experto	32
Capítulo 5 Implementación del Prototipo.....	33
5.1. DM.....	34
5.1.1. Corpus.....	34
5.1.2. Árboles de Argumentos.....	34
5.1.3. Reglas de Prolog	35
5.1.4. Conclusiones.....	36
5.1.5. Actualización.....	36
5.2. Sistema Experto	37
5.2.1. Módulo Cargar Instancia.....	37
5.2.2. Módulo Configuración	37
5.2.3. Módulo Diálogo	39
5.2.4. Módulo Aceptación/Rechazo de la Recomendación.....	41
5.3. Diagrama de Clases	42
Capítulo 6 Experimentación y Resultados.....	44

6.1. Condiciones de Experimentación.....	44
6.1.1. Sistema de Recomendación.....	44
6.1.2. Caracterización del Usuario en base al Sistema de Recomendación.....	45
6.1.3. Instancias.....	48
6.2. Resultados.....	51
6.2.1. Prueba Estadística.....	56
Capítulo 7 Conclusiones y Trabajos Futuros.....	58
7.1. Conclusiones.....	58
7.2. Aportaciones.....	59
7.3. Trabajos Futuros.....	59
Anexo A Ejemplo de Caracterización del Usuario.....	60
A.1. Texto Argumentativo.....	60
A.2. Árbol de Argumentos.....	60
A.3. Inserción en Base de Datos.....	61
A.4. Reglas de Prolog.....	62
A.5. Recomendación Inicial.....	63
A.6. Hechos.....	63
A.7. Conclusiones.....	64
A.8. Selección de Conclusiones.....	65
A.9. Actualización.....	66
Bibliografía.....	67

Índice de Figuras

Figura 1.1. Sistema de recomendación para la ayuda a la toma de decisión.....	3
Figura 1.2. Ejemplo de diálogo	4
Figura 1.3. Analogía del Sistema Experto y el Tomador de Decisiones	5
Figura 1.4. Colaboración de estudio del problema de Cartera de Proyectos Públicos.....	6
Figura 1.5. Arquitectura de trabajo	7
Figura 2.1. Frente de Pareto	15
Figura 2.2. Esquema de Soporte para ayuda a la Toma de Decisiones.....	20
Figura 2.3. Ejemplo de Árbol de Argumentos en Araucaria.	24
Figura 4.1. Diagrama General del procedimiento de Caracterización del Decisor y Diálogo	28
Figura 4.2. Diagrama Modular del Sistema Experto	32
Figura 5.1. Procedimiento de Caracterización del Decisor y diálogo con el Sistema Experto.....	33
Figura 5.2 Diagrama Entidad-Relación de la base de datos del CORPUS.....	35
Figura 5.3 Script de Salida de Prolog.....	36
Figura 5.4. Clase Instancia.....	37
Figura 5.5. Diagrama de flujo Premisas iniciales.	38
Figura 5.6. Ejemplo de Reglas de Locución.....	39
Figura 5.7. Ejemplo de Diagrama de Transición de Estados.....	39
Figura 5.8. Diagrama de flujo Estándar de Prueba.	39
Figura 5.9. Clase Recomendar Solución.	40
Figura 5.10. Clase Interacción.....	40
Figura 5.11. Clase Esquemas de Argumentación.	41
Figura 5.12. Diagrama de Clases.	43
Figura 6.1. Diagrama de Transición de Estados del Sistema de Recomendación.....	45
Figura 6.2. Ejemplo de Diálogo del Sistema de Recomendación.....	48
Figura 6.3. Resultados de la prueba estadística.....	56
Figura A.1. Texto argumentativo, transformado en Árbol de Argumentos..	61

Figura A.2. Representación de reglas en Prolog.....	62
Figura A.3. Ejemplo de carteras válidas generadas por el Sistema Experto.	63
Figura A.4. Ejemplo de recomendación generada por el Sistema Experto..	63
Figura A.5. Hechos obtenidos por parte del Sistema de Recomendación.	64
Figura A.6. Caracterización de reglas.	64
Figura A.7. Obtención de conclusiones.....	65
Figura A.8. Caracterización de las acciones Retar y pregunta crítica.....	65
Figura A.9. Selección de conclusiones por medio de la media aritmética.	66
Figura A.10. Actualización de hechos.....	66

Índice de Tablas

Tabla 1. Ejemplo sobre Características de relojes.....	4
Tabla 2. Resumen de las características sobre los trabajos del Estado del Arte	27
Tabla 3. Opciones de respuesta del Sistema de Recomendación	47
Tabla 4. Instancias de la Experimentación.....	50
Tabla 5. Información obtenida de las instancias.....	50
Tabla 6. Diálogos Simulados	51
Tabla 7. Información obtenida de diálogos simulados	53
Tabla 8. Comparación de Diálogos y discrepancias entre ellos.	54
Tabla 9. Comparación de Orden de Aparición	55

Capítulo 1. Introducción

1.1. Antecedentes

La toma de decisiones no es un proceso sencillo. Involucra una gran cantidad de variables, tantas que hacen complejo elegir una opción, debido a la alta cantidad de alternativas disponibles. Y, si esta acción es de importancia en la vida diaria, cuanto más lo será al involucrarse presupuestos y proyectos de empresas, grupos u organizaciones.

Por ello es que las ciencias de la computación siempre se encuentran a la vanguardia, estableciendo tendencias y otorgando aportaciones a las diferentes disciplinas que existen. Dentro del área de la toma de decisiones, las ciencias de la computación han establecido algoritmos y sistemas que la apoyan.

La finalidad de involucrar las ciencias de la computación, será el buscar la automatización de procesos, con la finalidad de apoyar y facilitar las diferentes tareas existentes y tener un mejor rendimiento.

En particular, se abordará el Problema de Cartera de Proyectos Públicos (PPP, por sus siglas en inglés), que se caracteriza por perseguir objetivos cuyo cumplimiento favorece a la sociedad. Dichos objetivos son generalmente intangibles, tal como la repercusión social y científica, así como la formación de recursos humanos, entre otros.

El DM es el encargado de valorar las acciones de política pública y sus consecuencias; la valoración es a través de su particular visión que está influenciada principalmente por sus juicios, creencias e interpretación de la realidad.

Por lo anterior, se puede concluir la importancia de generar procesos automatizados, ya que estos tendrán un impacto positivo en la toma de decisiones.

A continuación, se expresará con mayor detalle el contenido de los temas dentro de este trabajo de tesis, cuya aportación principal es la caracterización de la postura de un DM, lo cual representa un avance hacia la automatización mediante la interacción en un Sistema de Recomendación.

En este Capítulo, se abordarán los temas fundamentales de este trabajo de tesis. Está compuesto por la Definición del Problema, Justificación y beneficios, Planteamiento del Problema, Objetivos (General y Específicos), Alcances y Limitaciones, así como la Organización del documento.

1.2. Definición del Problema

En general, las instituciones, empresas y organizaciones tienen un cúmulo de proyectos que desean realizar, sin embargo tienen una cantidad limitada de recursos que se pueden invertir para dichos proyectos, lo cual significa que no se pueden apoyar todos éstos al mismo tiempo,

sino que es necesario seleccionar el conjunto al cual se va a dirigir la inversión de dichos recursos.

Esta situación recae en el DM, una persona o grupo encargado de decidir qué proyectos deben ser apoyados y cuáles no, sin embargo esta actividad es de mucha complejidad, ya que usualmente existe una cantidad tan grande de opciones que el razonamiento humano es insuficiente para analizarlas todas.

Por lo tanto, esto puede conllevar a una mala decisión, y derivar en repercusiones tales como:

- Desperdicio de recursos.- Los recursos son mal repartidos en los proyectos incorrectos, perdiendo beneficios potenciales.
- Pérdida de beneficios.- La empresa pierde los beneficios que les hubiera dado una elección correcta de proyectos a invertir.

En el caso del PPP, el tomar una decisión equivocada, puede hacer que los objetivos no se alcancen de la manera esperada (como puede ser beneficios al área de salud, educación, etcétera). El proceso de la toma de decisiones no es nada sencillo, ya que involucra múltiples objetivos que en diversas ocasiones son contrarios el uno con el otro.

Por ello, el objetivo de este trabajo de tesis es introducirse dentro de la toma de decisiones y los sistemas de recomendación, que son herramientas que le permiten a la persona o conjunto encargado de destinar los recursos a los proyectos, analizar las posibles soluciones de una manera más eficiente.

Esto debido a que el mismo sistema le ofrece los conjuntos de soluciones que buscan optimizar los resultados y ganancias y al mismo tiempo satisfacer cualquier restricción o preferencia establecida por el DM o la empresa.

Para que un sistema de recomendación funcione de una forma pertinente, una estrategia potencial que se puede emplear es la integración de teoría de argumentación y juegos de diálogo, entre el sistema y el decisor.

Es necesario implementar un proceso de argumentación que funcione de manera adecuada, esto con el objetivo de que el DM quede completamente satisfecho con la recomendación hecha por el sistema.

Este trabajo de tesis busca generar dicho proceso basándose en los esquemas de argumentación y caracterización, los cuales funcionan como una plantilla y permiten que el cliente sea capaz de entender las soluciones del sistema de una manera más fácil y sencilla.

La finalidad de este trabajo de investigación, es desarrollar un sistema de recomendación dentro del entorno de trabajo que sea capaz de recomendar, explicar y justificar al cliente una solución o un conjunto de posibles soluciones al PPP.

De esta manera, se ofrecerán siempre los mejores resultados posibles que cumplan con las restricciones establecidas (considerando que los objetivos están en conflicto en la mayoría de los casos) y con el conjunto de preferencias dado.

Cabe notar que la generación de dicho sistema no es tarea fácil, ya que los problemas de cartera de proyectos involucran criterios subjetivos. A dichos criterios, se les debe buscar asignar un valor real para su manipulación y posibles cambios dentro del proceso.

El valor real sirve para un cambio de preferencias del DM o el surgimiento de nueva información que conlleva a un conjunto de nuevas soluciones. En la Figura 1.1, se muestra de forma resumida un esquema de estrategias de apoyo a la decisión.

En ella, se puede ver como el DM, al necesitar tomar una decisión, se apoya en la técnica Métodos de Análisis de Decisión Multicriterio (llamada MCDA, por sus siglas en inglés). Sin embargo, para facilitar la toma de decisión, se utiliza un sistema de recomendación.

Dicho sistema, utiliza la teoría de la argumentación y los juegos de diálogo para su funcionamiento (Estos temas se desarrollan en las secciones 2.4.1, 2.4.2 y 2.4.3). Es en este sistema de recomendación, que se caracterizará al DM, con la finalidad de emular la postura que tendría al momento de tomar decisiones.

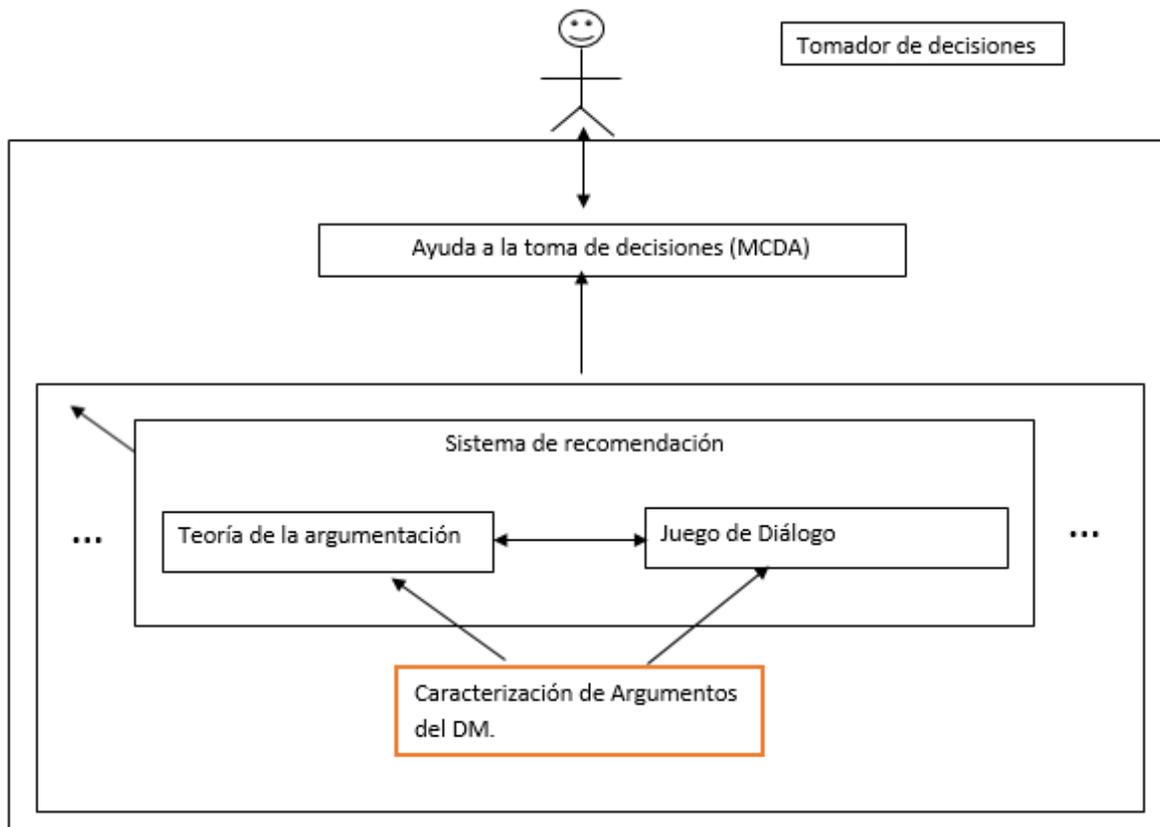


Figura 1.1. Sistema de recomendación para la ayuda a la toma de decisión.

Para poder entender este esquema de una mejor manera, se presenta el Ejemplo 1.1, que originalmente es presentado en el trabajo de Ouerdane (Ouerdane, 2006).

Ejemplo 1.1. Suponga que una clienta desea comprar un reloj. El problema radica en que una vez que ella se encuentra en la tienda de relojes, debe de tomar una decisión y seleccionar entre diferentes modelos, tamaños, precios y estilos de relojes, complicando la selección.

Entonces, el vendedor tiene una conversación con el cliente, para tratar de ubicar sus preferencias y así darle una respuesta más adecuada a lo que busca. Entonces, tras dicha conversación, el vendedor nota que tiene bien definidas las siguientes preferencias:

1. Prefiere un reloj pequeño en vez de uno mediano o grande.
2. Prefiere que el material del cual esté hecho sea acero y no de cuero.
3. En el color específico que prefiere el blanco, antes que el rojo o el rosa.
4. Le es más importante que el estilo del reloj sea moderno, antes que de tipo clásico o deportivo.
5. Además, busca que el modelo sea lo más barato posible.
6. Le interesa que el reloj tenga una garantía en caso de que se rompiera el reloj.

Con base a la información anterior, se seleccionaron 4 modelos de reloj los cuales se muestran en la Tabla 1.

Tabla 1. Ejemplo sobre Características de relojes.

Reloj	Tamaño	Material	Precio	Color	Estilo
A	Pequeño	Acero	450	Rojo	Clásico
B	Grande	Cuero	300	Blanco	Moderno
C	Mediano	Acero	320	Rosa	Clásico
D	Pequeño	Cuero	390	Rosa	Deportivo

Tras observar los diferentes modelos presentados por el vendedor, se desarrolla un diálogo entre el cliente y el vendedor, el cual puede observarse en la Figura 1.2. Ahí vemos claramente cómo el cliente cuestiona las recomendaciones del experto, lanzando preguntas críticas y argumentando con base a la información previa.

1. Experto: Dada tu información, **B** es la mejor opción
2. Cliente: ¿Por qué es este caso?
3. Experto: Porque **B** es globalmente mejor que las otras opciones.
4. Cliente: ¿Eso qué significa?
5. Experto: Bien... **B** está al máximo en la mayoría de los criterios considerados: el precio, el color, y especialmente el estilo, ¡es tan actual!
6. Cliente: Pero, ¿por qué **B** es mejor que **C** en el precio?
7. Experto: Porque **C** es 20 euros más caro que **B**.
8. Cliente: Estoy de acuerdo, pero yo veo que la garantía es muy cara, especialmente para este reloj. De hecho, no estoy seguro de querer la garantía.
9. Experto: Pero **C** es 5 euros mas caro que **B**.
10. Cliente: Lo noto, pero la diferencia no es significativa. Y también he cambiado de opinion: Yo debería tener un modelo clásico, pienso que es más conveniente para el uso diario.
11. Experto: Bien. En este caso, recomiendo a **C** como la mejor opción.
12. Cliente: ...

Figura 1.2. Ejemplo de diálogo.

La Figura 1.3, permite ver con mayor claridad el papel del experto y el cliente, pues realiza una analogía entre el experto (que viene siendo el sistema de recomendación) y el cliente (que es el corpus, realizado en este trabajo de investigación, y que representa al DM).

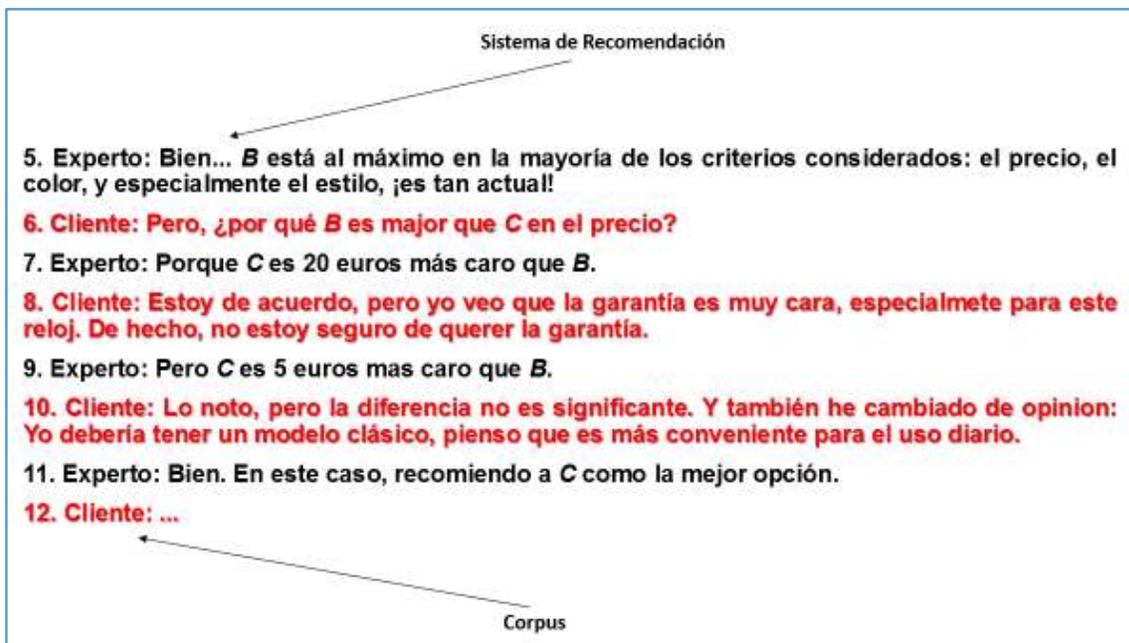


Figura 1.3. Analogía del Sistema Experto y el Tomador de Decisiones.

Todo esto nos permite observar que existe una postura del DM que puede ser detectada en los diálogos entre el cliente y el experto. Por tanto, es de suma importancia para crear un DM

artificial que se emule su postura con la mayor fidelidad posible para automatizar la interacción.

Este proyecto forma parte de un macro-proyecto que involucra a la Universidad Autónoma de Sinaloa (UAS), la Universidad Autónoma de Nuevo León (UANL) y el Instituto Tecnológico de Ciudad Madero (ITCM) con el objetivo de probar la posibilidad de una colaboración en conjunto (Esto se puede ver de manera más adecuada en la Figura 1.4).

El problema principal que se busca tratar en dicho macro-proyecto es la optimización de carteras de proyectos, lo cual requiere la aplicación de modelos y técnicas pertenecientes a las ciencias de la computación y las matemáticas.

La selección de carteras se ha definido en cuatro fases principales, donde la interacción con el usuario es de vital importancia en cada una de estas.

Las primeras dos etapas son la *captura de proyectos* y la *evaluación de proyectos*, estableciendo los objetivos a evaluar, las propuestas del proyecto y haciendo un análisis para definir la aportación y factibilidad de cada uno de los objetivos.

Después se llevan a cabo las fases de *optimización de carteras* y la *presentación de la recomendación*. En estas fases se buscan las mejores carteras de proyectos que satisfagan los criterios del DM. Dichas carteras se le muestran al DM para que defina sus preferencias con respecto a ellas.

El ITCM está apoyando el desarrollo de algoritmos de optimización para resolver el PPP, aprovechando la experiencia de sus miembros en la creación de algoritmos para la resolución de problemas de alta complejidad.

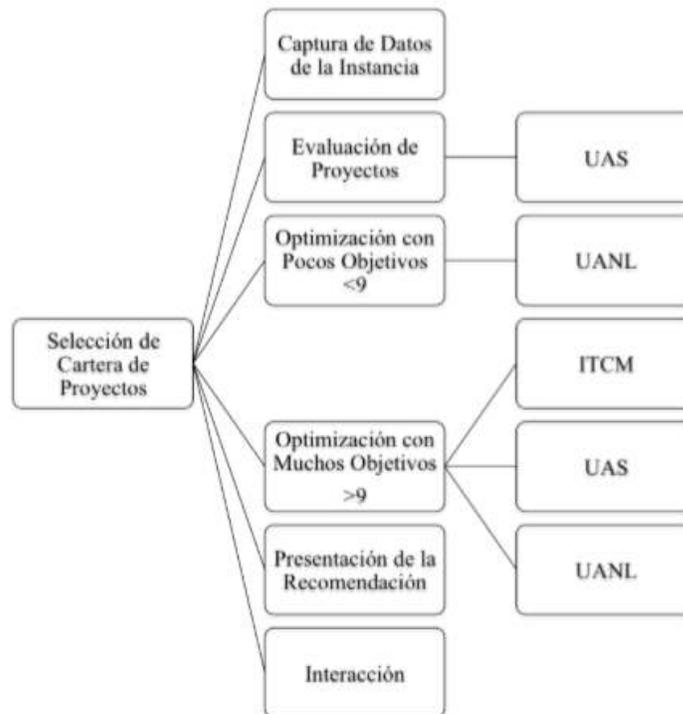


Figura 1.4. Colaboración de estudio del problema de Cartera de Proyectos Públicos.

1.3. Justificación y beneficios

La evaluación y selección de un conjunto de proyectos con impacto social es una de las principales tareas dentro de organizaciones gubernamentales, descentralizadas, fundaciones, grupos enfocados a la investigación y desarrollo, entre otras.

Además, existe una cantidad limitada de recursos que necesitan ser asignados a dichos proyectos de una manera que logren satisfacer la demanda y optimizar los beneficios.

Comúnmente, esto nos dirige a un proceso de toma de decisión, el cual recae en manos de una sola persona (o grupo de personas, según sea el caso), quien decidirá la cantidad de recursos a asignar a cada uno de los proyectos posibles.

Usualmente cuando este paso no recibe la importancia adecuada, se llegan a tomar decisiones de baja calidad que llevan a consecuencias muy dañinas, por tal motivo, es necesario llevar a cabo un estudio exhaustivo de los proyectos en cuestión, aunque hasta la actualidad, no han surgido verdaderos avances debido a la complejidad de este problema.

Por ello, estos motivos son suficientes para sustentar que es necesario contar con sistemas de apoyo a la decisión basados en las matemáticas y ciencias de la computación (sobre todo la inteligencia artificial).

Dichos sistemas permitirán a las personas a cargo de tomar las decisiones, contar con herramientas confiables, efectivas y robustas que hagan posible la distribución de recursos de la manera óptima.

Se busca reforzar el apoyo a la consolidación del macro-proyecto con una herramienta de software que sea capaz de utilizarse en casos reales, por ello, es necesario abordar el problema con las diferentes variantes y dimensiones que esto implica.

Una de estas variantes implica validar los sistemas de recomendación generados, lo cual se puede hacer mediante un corpus que caracterice los argumentos de un decisor. El flujo de interacción con usuarios reales se puede describir en 4 etapas principales, como se muestra en la Figura 1.5.

A través del desarrollo de nuevas estrategias para apoyo a la toma de decisiones en cada una de estas etapas se podrá hacer cada vez más simple dicha interacción con usuarios reales apoyándose de éstos métodos, específicamente para la etapa de generación de recomendaciones.

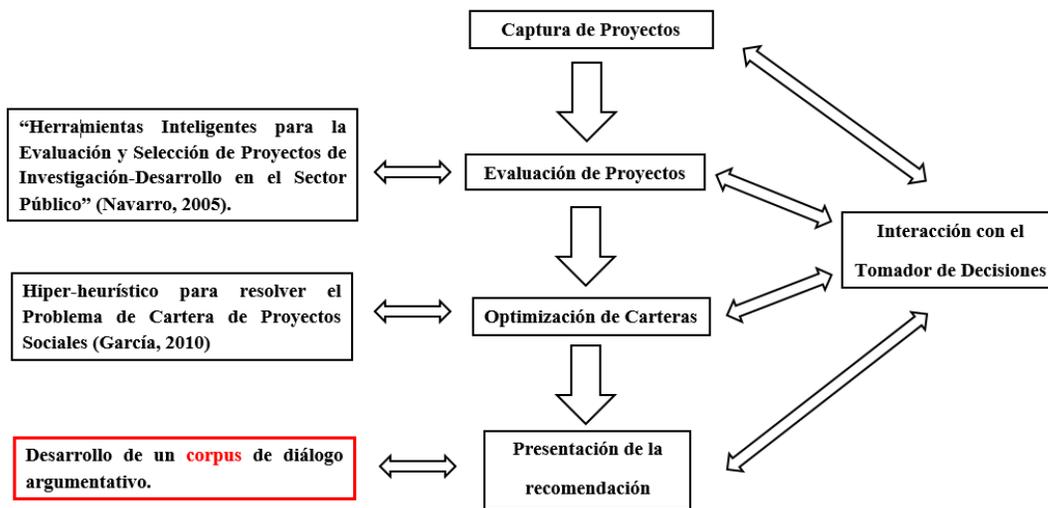


Figura 1.5. Flujo de Interacción con Usuarios Reales.

Con la planeación presentada con anterioridad, enfocada al PPP, las organizaciones públicas estarán capacitadas para repartir los recursos actuales dentro del entorno laboral, teniendo como consecuencia, la optimización de ganancias, satisfaciendo cada uno de los objetivos deseados, llevando a un incremento considerable en el beneficio para la sociedad.

Se tiene planeado abordar la caracterización de esquemas de diálogo argumentativo de un DM, con el objetivo de construir un corpus que sirva de base para validar un sistema de recomendación enfocado a las soluciones para el PPP.

Dentro de los objetivos se espera integrar aspectos teóricos con tecnologías emergentes, permitiendo la creación de una herramienta robusta y eficiente, la cual utilice métodos

formales para la solución de un problema real, que se encuentre validada lo mejor posible, para obtener mejores resultados.

1.4. Planteamiento del Problema

Dado un Tomador de Decisiones DM ; un Problema de Optimización Multiobjetivo P , y un Sistema de Recomendación R ,

¿Es posible desarrollar un corpus de diálogo argumentativo que caracterice el razonamiento del DM y valide las recomendaciones de R durante el proceso de solución de P ?

1.5. Objetivo

1.5.1. Objetivo general

Caracterizar los esquemas de diálogo argumentativo de un Tomador de Decisiones por medio de un corpus de diálogo argumentativo.

1.5.2. Objetivos específicos

- Análisis de Modelos para la Caracterización de Preferencias (o MCP) de un decisor
- Integrar un MCP en un prototipo del Sistema de Recomendación.
- Desarrollar un corpus de diálogos argumentativos (o CDA) que permita validar el Sistema de Recomendación.

1.6. Alcances y Limitaciones

Los alcances de este trabajo de tesis contemplan como objetos de estudio:

- Un problema de optimización. En el presente trabajo de investigación se abordará el Problema de PPP, el cual fue descrito en la Sección 3.3;
- Un modelo de caracterización de preferencias (corpus), cuyos conceptos fundamentales se expresan en la Sección 3.4; y
- Un sistema de recomendación basado en el corpus, el cual validará el funcionamiento correcto de este trabajo de tesis.

El sistema de recomendación de este trabajo de investigación está acotado por los siguientes aspectos:

- Sólo considerará instancias pequeñas del problema de optimización con un máximo de 3 objetivos y 25 proyectos;

- La administración del corpus se hará solo a través de una base de datos relacional;
- La interfaz de usuario se implementará en un ambiente local;
- No se considerará la evaluación del lenguaje natural dentro del proyecto; y
- No incluirá ningún esquema de reducción de argumentos.

1.7. Organización del documento

El trabajo de tesis está organizado de la siguiente manera:

En el Capítulo 2 se presentan conceptos fundamentales que apoyarán en el seguimiento al trabajo de tesis dentro del documento, en áreas como la toma de decisiones, el problema de PPP y el corpus.

El contenido del capítulo 3, habla de los trabajos relacionados con este trabajo de investigación. Se identificaron cuatro trabajos importantes que ayudan a dar un enfoque adecuado a este trabajo de investigación, ya que cada uno tiene diferentes elementos involucrados como ventajas y desventajas.

El capítulo 4 presenta la propuesta para implementar la caracterización del DM, por medio del corpus de diálogo argumentativo. Se hablarán de los dos elementos principales del prototipo, los cuales son el sistema de recomendación y el DM (artificial).

El capítulo 5 es la implementación en código de la propuesta. En esta sección se encuentran los diagramas de clases, entidad-relación, scripts de prolog y como está conformado el sistema de recomendación que interactúa con el DM artificial.

El capítulo 6 trata sobre la experimentación y resultados obtenidos con el sistema. Aquí se mencionan de las diferentes condiciones de experimentación (hardware y software), además de las instancias utilizadas. Los resultados son mostrados en tablas, con la finalidad de facilitar su comprensión.

En el capítulo 7 se presentan las conclusiones de la investigación realizada, así como las aportaciones principales y trabajos futuros.

Capítulo 2. Marco Teórico

Dentro de esta sección se definen los conceptos más importantes relacionados, con la definición del presente trabajo de investigación. Su relevancia es brindar el conocimiento teórico necesario para el desarrollo adecuado en el presente trabajo de tesis.

2.1. Toma de Decisiones

2.1.1. El problema de toma de decisión

Hablando en términos generales, tomar una decisión reside en seleccionar la mejor opción o alternativa de entre un conjunto de alternativas posibles. La incertidumbre suele ser una compañera presente en los procesos de toma de decisión que produce malestar e inseguridad a los individuos que deben tomar las decisiones (Sánchez, 2007).

La toma de decisiones con base en el trabajo de Keeney y Raiffa (Keeney y Raiffa, 1993) tiene como objetivo ayudar a los individuos a tomar decisiones difíciles y complejas de una forma racional.

Debido a dicha racionalidad, necesitó el desarrollo de métodos y/o modelos que permitieron plasmar de una forma fiel cada problema y analizar las diferentes alternativas con criterios objetivos.

Por tal motivo, se han integrado conceptos y métodos de disciplinas como la Estadística, la Economía y la Matemática, a la cual se le ha unido la Inteligencia Artificial, para desarrollar teorías y modelos en el campo de la toma de decisiones.

Estas han permitido estructurar, de forma lógica y racional, el proceso de toma de decisión y disminuir esta tarea a los individuos encargados de llevarla a cabo (Sánchez, 2007).

En cualquier problema de decisión, al encargado de tomar la decisión final es el DM. Este elemento central es una persona o grupo, cuyo sistema de preferencias es determinante en la solución de problemas que consideran varios objetivos, los cuales posiblemente se encuentren en conflicto entre sí (Fernández, 2011).

Las preferencias del DM van a tener una repercusión en el resultado final; pues como se ha mencionado con anterioridad, al haber conflicto en los objetivos, no existirá una solución óptima.

2.1.2. Métodos de Análisis de Decisión Multicriterio (MCDA)

El concepto de análisis es diferente en los problemas multiobjetivo. Para el PPP, no basta con comparar proyectos entre sí, ya que esto no está garantizado que el conjunto de los mejores proyectos sea la mejor cartera.

Por ejemplo, bajo el escenario de selección de cartera es posible rechazar un buen proyecto (en términos de impacto social), debido a que requiere un financiamiento excesivo que pudiera ser incompatible con las preferencias de algún DM que desea favorecer más proyectos.

Las preferencias del DM para formar una cartera, pueden ser representadas desde diferentes perspectivas, usando diferentes enfoques para llegar a la meta. Dichos enfoques dependen de quien sea el DM (Una sola persona, o un grupo heterogéneo), así como que tanto esfuerzo está dispuesto a invertir el DM en encontrar la solución al problema.

2.1.3. Modelo de Preferencias

El apoyo a la decisión está basado en la elaboración de la información preferencial. La idea básica en la metodología de ayuda a la decisión es que, dado un problema de decisión, recopilamos información preferencial del DM.

De esta manera que su sistema de valores está representado fielmente o críticamente construido y por lo tanto somos capaces de construir un modelo que, cuando se aplica, debe de hacer la recomendación de acción para el DM.

Modelado de preferencias

El modelado de preferencias tiene como objetivo construir relaciones de preferencia en un conjunto de acciones, conjunto A , que se evalúan con respecto a un criterio. La representación de las preferencias del DM sobre el conjunto A constituye un paso crucial en la ayuda a la decisión.

Dependiendo del contexto del problema, la naturaleza de la información que somos capaces de manejar y las expectativas del DM, diferentes situaciones pueden aparecer.

Estructuras de Preferencia

El concepto matemático de *relación binaria* es comúnmente usado como una representación formal de relaciones de preferencia (modelos) definida en un conjunto finito dado que dichas relaciones son el resultado de la comparación de dos elementos.

Por lo tanto, teniendo en cuenta que A es un conjunto finito, la comparación por pares de sus elementos pueden resultar en relaciones binarias indiferentes que tienen propiedades

diferentes. Decimos que tales relaciones construyen una estructura de preferencias, si cumplen ciertas condiciones.

Una estructura de preferencias (Ozturk et al., 2005) es un conjunto de relaciones binarias $\{S_1, \dots, S_m\}$ definida en el conjunto A tal que:

- $\forall x, y \in A \exists i \in \{1, \dots, m\}, xS_i y$ o $yS_i x$. Significa para cada pareja x, y en A ; al menos una relación se cumple; y
- $\forall x, y \in A, xS_i y \Rightarrow \forall j \neq i, \text{ no } (xS_j y) \text{ y no } (yS_j x)$, lo que significa que para cada pareja x, y en A , si una relación es satisfecha, otra no puede ser satisfecha.

Una relación binaria grande, también llamado una relación de *outranking*, denotado por S , se puede utilizar para caracterizar una estructura de preferencia. Esta relación se interpreta como "a es al menos tan buena como b", denotado como $a \succeq b$ o aSb , tal que $\succeq \subset A \times A$.

Las reglas de preferencia del DM no están bien definidas. La existencia de zonas de incertidumbre en la mente del DM puede deberse a creencias imprecisas, conflictos o aspiraciones que se oponen entre sí (Roy, 1990).

Roy (Roy 1996) describe situaciones donde se presenta este comportamiento no ideal en el DM.

Debido a esto usa un sistema preferencial compuesto de varias relaciones binarias. A continuación se describen dichas relaciones:

- 1) *Indiferencia*: Corresponde a la existencia de razones claras que justifican una equivalencia entre las dos alternativas. Se denota como xIy .
- 2) *Preferencia estricta*: Corresponde a la existencia de razones claras que justifican una preferencia significativa en favor de alguna de las dos alternativas. La declaración "x es estrictamente preferida sobre y" se denota como xPy .
- 3) *Preferencia débil*: Corresponde a la existencia de razones claras para preferir a x sobre y, pero que no son lo suficientemente significativas como para justificar una preferencia estricta. Se utiliza cuando la indiferencia y la preferencia estricta no pueden ser claramente distinguidas. Es denotada como xQy .
- 4) *Incomparabilidad*: Ninguna de las situaciones anteriores predomina. Esto es porque faltan razones claras que justifiquen cualquiera de las relaciones anteriores. Se denota como xRy .
- 5) *Sobreclasificación*: Corresponde a la existencia de razones que justifican la declaración "x es al menos tan bueno como y", pero sin una división claramente establecida entre preferencia estricta, preferencia débil o indiferencia. La sobreclasificación se denota como xSy .
- 6) *k-Preferencia*: Corresponde a la existencia de razones claras que justifican ya sea una preferencia estricta o una incomparabilidad, pero sin existir una diferencia establecida entre ambas situaciones. Se denota como xKy .

- 7) *No preferencia*: Corresponde a la situación en la cual tanto la indiferencia como la incomparabilidad son posibles, sin existir una diferencia clara entre ellas. Se denota como $x \sim y$.

2.2. Optimización Multiobjetivo

2.2.1. Optimalidad de Pareto

Una gran parte de los problemas de decisión del mundo real implican el procesamiento simultáneo de múltiples criterios (Pérez, 2007). Los criterios, también llamados objetivos, con frecuencia están en colisión, es decir, al mejorar uno de los objetivos, los demás objetivos se pueden ver afectados (en forma negativa).

El término optimización, según Kalyanmoy Deb (Deb, 2001), es la acción de encontrar una o más soluciones factibles que corresponden a valores extremos de uno o más objetivos. Cuando un problema de optimización comprende solamente una función objetivo, se le llama optimización mono-objetivo, cuya meta es encontrar la solución óptima.

Por el contrario, cuando nos encontramos con un problema de optimización, donde se involucra más de una función objetivo, se le conoce como optimización multiobjetivo, cuya meta es la de encontrar una o más soluciones óptimas.

En un problema de optimización multiobjetivo (MOP por sus siglas en inglés (Multiobjective Optimization Problems) se busca un conjunto de soluciones que presenten un equilibrio óptimo entre los diferentes objetivos.

A la gráfica de éste conjunto de soluciones-equilibradas presentadas en el espacio de objetivos se le denomina frente de Pareto, por esta razón, estas soluciones son conocidas como soluciones óptimas de Pareto o soluciones no-dominadas (Van Veldhuizen y Lamont, 1998).

En problemas multiobjetivo, es muy común que los objetivos se encuentren en conflicto entre sí. Por ello, se buscan normalmente *soluciones compromiso* en vez de una única solución. Debido a esto, la noción de óptimo es diferente en estos casos.

La noción de óptimo más comúnmente aceptada es la que propuso Francis Ysidro Edgeworth (Edgeworth, 1881), la cual más tarde fue generalizada por Vilfredo Pareto (Pareto, 1896). Aquella noción es comúnmente llamada con el término de *optimalidad de Pareto*.

Se dice que un vector de variables de decisión: $\vec{x}^* \in \mathcal{F}$ es un óptimo de Pareto existente en la región factible \mathcal{F} si (asumiendo minimización):

$$f_i(\vec{x}^*) \leq f_i(\vec{x}), \forall i \in [1, 2, \dots, k] \quad \vee \quad f_j(\vec{x}^*) < f_j(\vec{x}), \exists j \in [1, 2, \dots, k] \quad (1)$$

En palabras, \vec{x}^* es un óptimo de Pareto si no existe otro vector factible de variables de decisión $\vec{x} \in \mathcal{F}$ que mejore algún criterio sin causar el deterioro simultáneo de al menos otro criterio.

Sin embargo, este concepto casi siempre ofrece no una, sino varias soluciones llamadas *Conjunto de óptimos de Pareto* (P^*). Los vectores \vec{x}^* correspondientes a las soluciones incluidas en el conjunto de óptimos de Pareto son llamados *no-dominados*. La imagen del conjunto de óptimos de Pareto bajo las funciones objetivo es llamada frente de Pareto.

En algoritmos multiobjetivo, se utiliza con mucha frecuencia el concepto de *dominancia de Pareto* al contrastar dos soluciones y decidir si una domina a otra o no. Se dice que una solución domina a otra si se cumplen las siguientes condiciones (para el caso de minimización):

- 1) La solución \vec{x}_a no es peor que \vec{x}_b en todos los objetivos, o:

$$f_i(\vec{x}_a) \leq f_j(\vec{x}_b), \quad \forall i \in [1, 2, \dots, k] \text{ y}$$
- 2) La solución \vec{x}_a es estrictamente mejor que \vec{x}_b en al menos un objetivo, o:

$$f_i(\vec{x}_a) < f_j(\vec{x}_b), \quad \exists i \in [1, 2, \dots, k] \text{ y}$$

Si alguna de las condiciones no se cumple, entonces se dice que la solución no domina a la solución. En otras palabras, para que una solución domine a otra, es necesario que sea estrictamente mejor en al menos un objetivo, y no peor en ninguno de ellos. Por ello, al comparar dos soluciones A y B, sólo pueden existir tres posibles soluciones:

- A domina a B
- A es dominada por B
- A y B no se dominan (son no dominadas entre sí)

La representación de las funciones objetivo cuyos vectores son no-dominados y además están en el conjunto de óptimos de Pareto es llamado el Frente de Pareto.

Para un problema multiobjetivo planteado y un conjunto de óptimos de Pareto, el frente de Pareto se define como:

$$\mathcal{FP}^* = \{F(\vec{x}) = (f_1(\vec{x}), f_2(\vec{x}), \dots, f_k(\vec{x})) \mid \vec{x} \in P^*\}$$

Hablando en general, no existe un método que sea completamente eficiente para encontrar el frente de Pareto, y la mejor forma de realizarlo, es hacer pruebas con todos y cada uno de los puntos en la zona factible

Por lógica, el espacio de búsqueda es demasiado grande, por lo cual este proceso es incosteable y de ahí surge la necesidad de implementar heurísticas para generar aproximaciones del frente de Pareto.

A continuación, en la Figura 2.1, se muestra de forma gráfica, el frente de Pareto, tanto para problemas de minimización como de optimización.

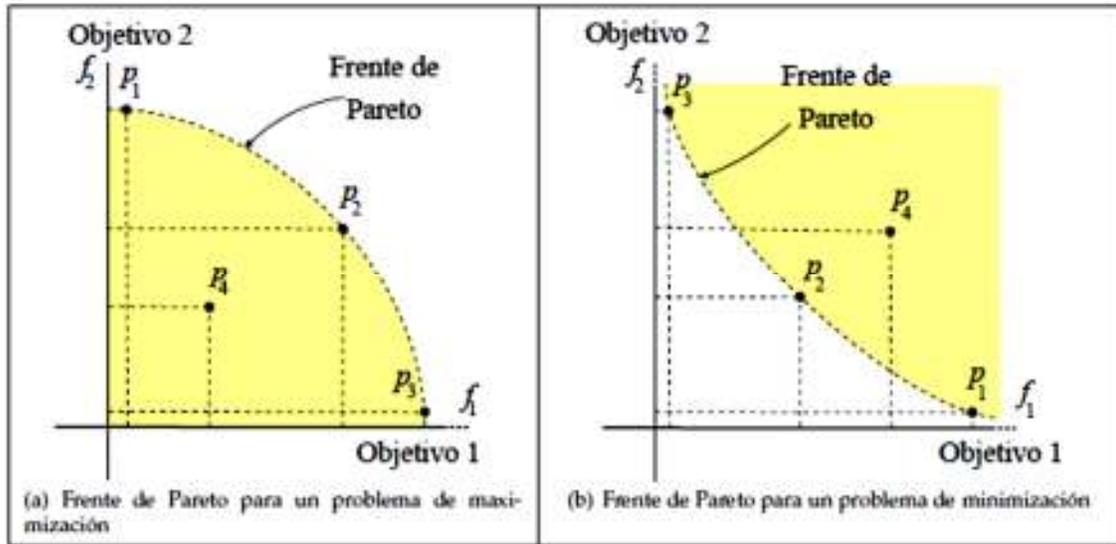


Figura 2.1. Frente de Pareto

2.2.2. Modelo Matemático de un Problema Multiobjetivo

Formalmente, en un problema multiobjetivo se busca el vector de variables de decisión \vec{x} , el cual se representa a continuación.

$$\vec{x} = [x_1, x_2, \dots, x_n]^T$$

Este vector tiene el objetivo de optimizar a $F(\vec{x})$ (Serrano, 2007). Por lo tanto, el modelo matemático quedaría de la siguiente manera:

$$F(\vec{x}) = [f_1(\vec{x}), f_2(\vec{x}), \dots, f_k(\vec{x})], f_i: \mathbb{R}^n \rightarrow \mathbb{R} \quad (2)$$

Sujeto a:

$$g_i(\vec{x}) \leq 0; i = 1, 2, \dots, m;$$

$$h_j(\vec{x}) = 0; j = 1, 2, \dots, p$$

Donde:

k es el número de funciones objetivo

n es el número de variables de decisión

m es el número de restricciones de desigualdad

p es el número de restricciones de igualdad.

En otras palabras, se busca determinar el conjunto de todos aquellos números que satisfagan las restricciones y que optimicen todas las funciones objetivo. Las restricciones definen la región factible del problema y todo vector que se encuentre en esta región será considerado como una solución factible (Santana, 2004). Los problemas multiobjetivo son de tres tipos:

- Minimizar todas las funciones objetivo
- Maximizar todas las funciones objetivo
- Minimizar algunas y maximizar las funciones objetivo restantes

Para abordar la solución de MOPs, existen algoritmos exactos y aproximados. Dentro de los algoritmos aproximados, se encuentran estrategias deterministas y estocásticas. Una de las estrategias estocásticas de mayor popularidad los constituyen los algoritmos evolutivos, los cuales serán base de esta investigación.

2.2.3. Estrategias de Optimización Multiobjetivo

La optimización estocástica es una clase general de algoritmos que utiliza algún grado de aleatoriedad para conseguir soluciones óptimas (o las más cercanas posibles) a problemas de optimización difíciles.

Las metaheurísticas son la clase más general de este tipo de algoritmos, algunos ejemplos muy utilizados en la literatura son el recocido simulado, búsqueda tabú, optimización mediante colonia de hormigas y algoritmos evolutivos (Luke, 2009).

Sin embargo, debido a la forma en que trabajan dichos algoritmos, las soluciones que generan no siempre son las mejores posibles y, en general, su optimalidad no puede ser respaldada (Serrano, 2007).

Aunque a simple vista, esto pareciera otorgar una gran desventaja al respecto, muchos resultados experimentales han mostrado que una metaheurística bien diseñada, tendrá una gran probabilidad de proporcionar soluciones muy cercanas a las óptimas en tiempos de cómputo sensatos (Gendreau y Potvin, 2005).

Metaheurísticas evolutivas

Dentro de las metaheurísticas se encuentran los algoritmos evolutivos, que son una clase de métodos de optimización estocástica que simulan el proceso de la evolución natural, entre las metodologías propuestas destacan los algoritmos genéticos (Zitzler, 1999).

Los algoritmos evolutivos se caracterizan por contar con una población de posibles soluciones y un proceso de reproducción, el cual permite la combinación de soluciones existentes para generar nuevas soluciones. Cuentan con una técnica de selección, la cual determina qué individuos de la población en turno participan en la nueva población.

Este proceso permite encontrar varios miembros del conjunto de óptimos de Pareto en una sola ejecución del algoritmo, en lugar de tener que realizar una serie de ejecuciones por separado, como ocurre con los algoritmos clásicos y con técnicas estocásticas no poblacionales (p.ej., el Recocido Simulado) (Serrano, 2007).

De acuerdo a (Goldberg, 1989), las principales ventajas que presenta el uso de los algoritmos evolutivos en la resolución de problemas de optimización son entre otras, las siguientes:

- Operan sobre una población (o conjunto de soluciones) lo que evita que la búsqueda se quede atascada en óptimos locales.
- No requieren conocimiento previo sobre el problema a resolverse.
- Pueden combinarse con otras técnicas de búsqueda para mejorar su desempeño.
- Permiten su paralelización de forma sencilla.
- Son conceptualmente fáciles de implementar y usarse.

Algoritmos evolutivos multiobjetivo (MOEAs)

Los algoritmos evolutivos para optimización multiobjetivo (MOEAs) por sus siglas en inglés (Multiobjective Evolutionary Algorithms) han mostrado su potencial para alcanzar la meta trazada por la comunidad de optimización multiobjetivo: aproximación bien distribuida y buena convergencia (Laumanns et al., 2002; Li, 2012; Tan, 2008).

Sin embargo, para fines prácticos, el DM no está interesado en que se le presente todo el frente de Pareto, sólo quiere una solución que corresponda con sus preferencias.

Con la finalidad de encontrar esa única solución, nuevos métodos, que en su mayoría son variantes de los MOEAs existentes, utilizan las preferencias del DM para guiar la búsqueda hacia una porción preferida del frente, por ejemplo la región de interés (Region of Interest, ROI) definida en (Adra et al., 2007).

Algoritmos evolutivos multiobjetivo basados en preferencias

Se mencionarán algunas de las técnicas evolutivas que se han propuesto para incorporar preferencias en optimización multiobjetivo, dividiéndolas según la clasificación propuesta por Cohon y Marks (Cohon y Marks, 1975).

Técnicas a priori: Las preferencias del usuario tienen que ser conocidas antes de que comience la búsqueda.

Técnicas progresivas: En estas técnicas, las preferencias se van dando conforme la búsqueda avanza y el DM indica si una solución le parece adecuada o no y el proceso actualiza las preferencias conforme el DM lo va indicando, guiando así el proceso de búsqueda.

Técnicas a posteriori: En estas técnicas, las preferencias se expresan al final y el DM recibe una información completa de los resultados para así entonces tomar la decisión que mejor le convenga.

En otras palabras, los resultados intentan mostrar todos los compromisos posibles entre las funciones objetivo tratando de generar el verdadero frente de Pareto o al menos una aproximación razonablemente buena.

2.3. Problema de Cartera de Proyectos Públicos

Un problema particular de optimización multiobjetivo es el PPP, el cual consiste en la selección de los proyectos públicos que más beneficios provean a la sociedad, por ejemplo, proyectos abocados a educación, salud y transporte público.

Debido a su naturaleza, el problema de selección de proyectos ha sido comúnmente abordado mediante algoritmos multiobjetivo (Carazo et al., 2010; Doerner et al., 2004; Reiter, 2010) que se enfocan a identificar un conjunto de soluciones en el frente de Pareto.

2.3.1. Proyecto

Un proyecto es un proceso temporal, único e irrepetible que persigue un conjunto específico de objetivos (Carazo et al., 2010). Por otro lado, una cartera de proyectos es un conjunto de proyectos realizados en el mismo lapso de tiempo (Carazo et al., 2010).

Por esta razón, los proyectos que están en una misma cartera, comparten los recursos disponibles en la organización, además, pueden complementarse entre ellos.

Con ello, podemos establecer que no es suficiente comparar los proyectos de manera individual, sino que se deben comparar grupos de proyectos para poder identificar cuál cartera realiza una aportación mayor a los objetivos de la organización.

La correcta selección de proyectos para integrar la cartera, en la cual se invertirán los recursos de la organización, es uno de los problemas más importantes de decisión tanto para instituciones públicas como privadas (Castro, 2007; García, 2010).

2.3.2. Características del Problema de Cartera de Proyectos Públicos

En el PPP, el DM es el encargado de valorar las acciones de política pública y sus consecuencias; la valoración es a través de su particular visión que está influenciada principalmente por sus juicios, creencias e interpretación de la realidad.

Por la razón mencionada con anterioridad, se considera que el DM es el factor subjetivo del problema de decisión (Fernández et al., 2011).

En particular, los proyectos públicos se caracterizan por perseguir objetivos cuyo cumplimiento favorece a la sociedad. Dichos objetivos son generalmente intangibles, tal

como la repercusión social y científica, así como la formación de recursos humanos, entre otros, dejando de lado el posible beneficio económico como elemento principal de medida.

La selección de proyectos de una cartera de proyectos públicos necesita de un tratamiento especial por las siguientes razones (Fernández y Navarro, 2002):

- 1) La calidad de los proyectos es generalmente descrita por múltiples criterios que frecuentemente están en conflicto.
- 2) Comúnmente los requerimientos no son conocidos con exactitud. Muchos conceptos no tienen un soporte matemático por ser de naturaleza totalmente subjetiva.
- 3) La heterogeneidad, o diferencia entre los objetivos perseguidos por los proyectos, dificulta compararlos.

El problema que es el objeto de estudio en esta investigación es el problema de cartera de proyectos, cuya definición formal se presenta a continuación.

Consideremos una cartera de proyectos a desarrollar de los cuales el DM debe elegir algunos proyectos de dicho conjunto con el objetivo de obtener el mayor grado de beneficios, sin excederse de los recursos establecidos.

Considérese un vector p -dimensional $f(i) = \langle f_1(i), f_2(i), f_3(i), \dots, f_p(i) \rangle$, donde cada $f_j(i)$ representa la contribución del proyecto i para el objetivo j , por otra parte el portafolio es comúnmente modelado como un vector binario $x = \langle x_1, x_2, \dots, x_n \rangle$ donde $x_i = 1$ si el proyecto i es apoyado y $x_i = 0$ en caso contrario.

Teniendo un límite de recursos B y un costo c para cada proyecto, la restricción de presupuesto se puede establecer de la siguiente manera:

$$\left(\sum_{i=1}^N x_i c_i \right) \leq B \quad (3)$$

Además de la restricción de presupuesto, también deben tenerse en cuenta las restricciones por área, en otras palabras, que rangos máximos (L) y mínimos (U) por proyecto k de la cantidad total del presupuesto se puede destinar a dicha área, para eso se formula la siguiente restricción:

$$L_k \leq \sum_{i=1}^N x_i g_i(k) c_i \leq U_k . \quad (4)$$

Donde g está definido por:

$$g_i(k) = \begin{cases} 1 & \text{if } a_i = k, \\ 0 & \text{otherwise} \end{cases} \quad (5)$$

De igual manera, cada proyecto corresponde a un región geográfica que beneficiara, a estas regiones también se les aplica rangos máximos y mínimos y estas mismas restricciones.

Se puede observar que la solución al problema de cartera de proyectos requiere la toma de decisiones, la cual se puede abordar a través de la teoría de la argumentación, concepto que se aborda en la siguiente sección.

2.4. Teoría de Argumentación en la Toma de Decisiones

Un problema básico que se debe resolver dentro de la toma de decisiones es que, al momento de emitir una recomendación para el DM, se debe contar con información suficiente que la justifique.

Dentro del área de inteligencia artificial, los *sistemas de recomendación* representan el área de estudio para estas justificaciones, y son apoyadas comúnmente por diversas teorías, siendo de interés para esta investigación las basadas en *teoría de la argumentación*.

Los principales elementos involucrados en un *sistema de Recomendación* son: a) Sistemas de recomendación; b) Argumentos; c) Teoría de la argumentación; d) Juego de diálogos; e) Caracterización de argumentos un DM (enfocada en el PPP).

Estos son descritos brevemente en el resto de esta sección. La jerarquía del esquema de soporte entre los componentes para ayuda a la toma de decisiones se puede ver resumida en la Figura 2.2.

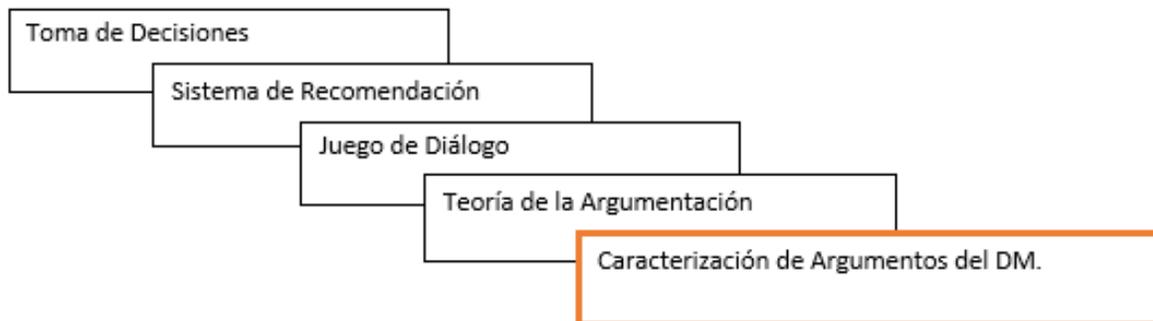


Figura 2.2. Esquema de Soporte para ayuda a la Toma de Decisiones.

2.4.1. Sistema de Recomendación

Un analista novato (o no experto) puede encontrar útil el tener el soporte de una herramienta que le provea con una explicación explícita, una justificación y posibles respuestas que puedan ocurrir en el curso de la interacción con su cliente.

En el caso de una herramienta automática, puede ser satisfactorio el permitir algún tipo de interacción con el cliente que le permita el refinar recomendaciones.

La ayuda a la decisión involucra el entender, interpretar, justificar, explicar, convencer, revisar y actualizar los resultados del proceso de ayuda a la decisión.

El proceso de ayuda a la decisión puede ser visto como una secuencia de artefactos cognitivos, producidos a través de la interacción entre el cliente y el analista.

Los cuatro artefactos cognitivos sugeridos por Tsoukiàs (Tsoukiàs, 2011) son los siguientes: 1) Una representación de la situación del problema; 2) Una formulación del problema; 3) Un modelo de evaluación; 4) Una recomendación final.

Emitir una recomendación es el último paso del proceso de ayuda a la decisión. El núcleo principal de la recomendación se basa en la argumentación.

La argumentación ofrece una alternativa para mecanizar razonamiento *no-monotónico*, es decir, razonamiento que puede cambiar el valor de verdad de algunas conclusiones cuando la información original es modificada.

Los frameworks basados en argumentos ven este problema como un proceso en el que los argumentos a favor y en contra de las conclusiones se construyen y se comparan.

La no-monotonicidad surge del hecho de que las nuevas premisas pueden permitir la construcción de nuevos argumentos para apoyar la nueva conclusión nueva o fuertes contra-argumentos en contra de las conclusiones existentes.

En un diálogo está presente la *no-monotonicidad* del razonamiento, implicando que las verdades pueden cambiar durante él. Sin embargo, en el proceso de automatización de apoyo a la toma de Decisión, una estrategia para lidiar con la no-monotonicidad es el uso de la *Teoría de la Argumentación*.

2.4.2. Teoría de la Argumentación

La teoría de la argumentación es un campo cada vez mayor de la Inteligencia Artificial. En pocas palabras, es el proceso de construir y evaluar argumentos para justificar las conclusiones.

Es un proceso para realizar decisiones prácticas justificadas, provee un mecanismo de razonamiento no monotónico, la posibilidad de cambiar conclusiones a la luz de nueva información (Ouerdane, 2006).

Los esquemas de razonamiento son formas de argumentos que capturan patrones estereotipados de razonamiento humano, especialmente anulables (Walton, 2005; Norman et al., 2003.).

Los argumentos son presentados como reglas de inferencia generales por lo que dado un conjunto de premisas, se puede llegar a una conclusión (Walton, 1996). Sin embargo, tales esquemas no son deductivamente estrictos debido a la naturaleza anulable de los argumentos.

Dentro de la argumentación, son muy importantes los esquemas. Dichos esquemas permiten a los argumentos ser representados dentro de un contexto particular y tomar en cuenta que el razonamiento presentado puede ser modificado a la luz de nuevas pruebas o excepción a las reglas.

Los esquemas de argumentación identifican las premisas y la conclusión del argumento, Existen diferentes tipos de premisas (Walton, 1996):

- *Premisas ordinarias*: Aquellas que deben ser apoyadas con más evidencia.
- *Suposiciones*: Aquellas que pueden ser asumidas hasta que han sido cuestionadas.
- *Excepciones*: Aquellas que no se mantienen en la ausencia de evidencia de lo contrario.

Según (Walton, 1996), los esquemas de argumentos pueden jugar dos roles:

- Cuando construyen argumentos, proveen un repertorio de formas de argumentos para ser considerados, y una plantilla para las piezas que se necesitan;
- Cuando se ataca, los argumentos proveen un conjunto de preguntas críticas que pueden identificar debilidades potenciales en el caso del oponente

Una forma de definir la lógica argumentativa está en la forma dialéctica de los *juegos diálogo* Tales juegos modelan la interacción entre dos o más jugadores, donde los argumentos a favor y en contra de una proposición se intercambian de acuerdo con ciertas reglas y condiciones (Carlson, 1983). Este concepto se describe en la siguiente sección.

2.4.3. Juegos de Diálogo

Los juegos de diálogo (o sistemas de diálogo) esencialmente definen el principio del diálogo coherente y la condición bajo la cual una declaración hecha por un individuo es adecuada.

Existen diferentes diálogos formales, teniendo en cuenta información variada, tal como: los participantes, el lenguaje de comunicación, los roles de los participantes, el objetivo del diálogo, etc.

Regla de locución (actos de habla, movimientos). Las reglas que indican qué expresiones son permitidas.

Por lo general, locuciones legales permiten a los participantes afirmar proposiciones, permite a otros cuestionar o impugnar las afirmaciones anteriores, y permitir que aquellas proposiciones de afirmación que son posteriormente cuestionadas o impugnadas al justificar sus afirmaciones.

Las justificaciones pueden involucrar la presentación de una prueba de la proposición o un argumento para ello.

Reglas de compromisos. Reglas que definen el efecto de los movimientos en los "compromisos". De hecho, asociados con cada jugador es un compromiso, que mantiene las declaraciones que los jugadores han hecho y los desafíos que hayan expedido. Existen, pues, las reglas que definen cómo los compromisos se actualizan.

Reglas de dialogo. Reglas para regular los movimientos. Se especifica, por ejemplo, el conjunto de los actos de habla permitidos en un diálogo y sus tipos de respuestas permitidos. Diversos protocolos de diálogo pueden ser encontrados en la literatura, especialmente para la persuasión (Prakken, 2001) y la negociación (Parsons et al, 1998;.. Amgoud et al, 2000b)

Reglas de terminación. Las reglas que definen las circunstancias en las que el diálogo termina.

Aceptabilidad. En un proceso de argumentación, es importante definir el estado de argumentos (o evaluarlos) sobre la base de todas las formas en que interactúan. Así, los argumentos mejores o aceptables deben ser identificados al final del proceso de argumentación.

La mayoría de los sistemas de argumentación están basados en la noción de aceptabilidad como se identificó por (Dung, 1995). Dung ha propuesto un framework abstracto para la argumentación en la que se centra sólo en la definición del estado de los argumentos.

En tal framework, la aceptabilidad de un argumento depende de su pertenencia a algunos conjuntos, llamados *conjuntos aceptables o extensiones*. En otros términos, la aceptabilidad de los argumentos se define sin considerar la estructura interna de los argumentos.

2.5. Corpus de Diálogo Argumentativo

2.5.1 Definición

De acuerdo a su definición en español (Larousse, 2015), se define *corpus* como un conjunto extenso de datos, textos, frases, etc., que pueden servir de base a una investigación.

Con base en los trabajos de Chotimongkol (Chotimongkol, 2008), Biran (Biran, 2011) y Benz Müller (Benz Müller, 2010) se puede definir un *corpus de diálogo* como un conjunto de textos, datos o frases que caracterizan el dialogo entre el sistema de recomendación y el Decision Maker.

Estas frases se encargan de respaldar los argumentos de las recomendaciones que el sistema otorgará.

2.5.2 Modelo de Toulmin

El trabajo de Toulmin (Toulmin, 1958) establece que los argumentos se pueden descomponer en 5 diferentes tipos de enunciados que pueden identificarse en los diferentes textos existentes y representarse visualmente o mapearse. Estos elementos son:

- Reclamo.- Es un enunciado o argumento en forma de oración declarativa.
- Aseveración.- Un argumento simple, que consiste de un reclamo y una razón/premisa.
- Premisa/Razón.- Cualquier oración que soporte un reclamo.
- Co-premisa.- Premisas que trabajan juntas para respaldar un reclamo o una aseveración.
- Objeción.- Cualquier oración que provea evidencia contra un reclamo o una aseveración.

2.5.3 Mapeo de Argumentos

En Botley (Botley, 2014) se implementa una estrategia llamada Mapeo de Argumentos (del vocablo inglés Argument Mapping).

Es una técnica de visualización, con la cual varios aspectos del proceso de razonamiento en discurso, pueden ser mapeados en una representación gráfica clara, que revela detalles interesantes sobre la persona, evidenciándola por medio del lenguaje que se utiliza.

Utiliza diagramas donde utiliza flechas para mostrar dependencias o relaciones entre el argumento principal (claim), junto con las premisas y objeciones que soportan a la idea central. Este mapa lo llamaremos árbol de argumentos, el cual tendrá importancia en la propuesta de solución (Capítulo 4).

Un ejemplo de software que utiliza la técnica de Mapeo de Argumentos, es Araucaria (Reed, 2004). En Moens (Moens, 2007) explica a grandes rasgos su funcionamiento.

En dicho artículo, se establece que es un software computacional que permite la generación gráfica de argumentos y sus relaciones. Es importante recalcar que este proceso se hace manualmente y no automáticamente.

Araucaria permite la creación de un árbol de argumentos, por medio de un grafo. El usuario selecciona fragmentos de un texto, para crear nodos, los cuales pueden relacionarse.

En la Figura 2.3, podemos observar un ejemplo de la técnica Argument Mapping, realizado en el software Araucaria. Del lado izquierdo se encuentra el texto argumentativo.

Posteriormente, el usuario identificará los diferentes tipos de argumentos y construirá un Árbol de Argumentos, el cual estará enlazado con las objeciones y premisas que soportan a la idea principal, permitiendo tener un mejor análisis al poder visualizar las ideas.

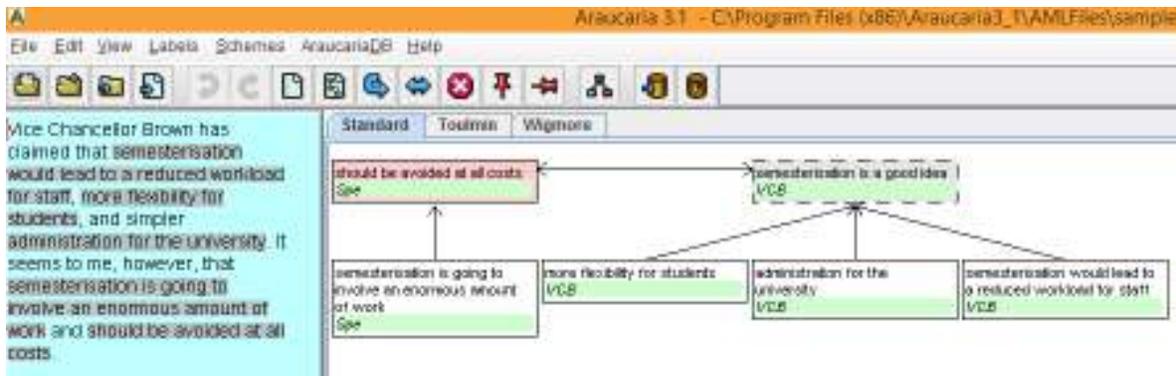


Figura 2.3. Ejemplo de Árbol de Argumentos en Araucaria.

Capítulo 3. Estado del Arte

Se llevó a cabo una revisión del estado del arte, tomando el caso particular de la caracterización del diálogo. Se identificaron cuatro trabajos importantes que guían sobre este tema.

Estos son el trabajo de (Chotimongkol, 2008), el trabajo de Hunter (Hunter, 2004), el trabajo de (Ouerdane, 2006), y el trabajo de (Cruz-Reyes, 2014). Dichos trabajos se describen a continuación.

3.1 Aprendizaje de la estructura de conversaciones orientadas a tareas desde un Corpus de diálogo de dominio específico.

En el trabajo de (Chotimongkol, 2008) se aborda el problema de desarrollar un sistema de diálogo en dominios orientados a tareas. Para la solución del problema propone una metodología en dos etapas.

En la primer etapa propone crear una representación del diálogo que encaje con información específica de dominio, esto lo lleva a cabo por medio de una caracterización sencilla del diálogo a través de una estructura de tres niveles, observada en humanos, definida por tarea, subtarea y concepto.

La segunda etapa trata de adquirir la estructura específica del diálogo por medio de la identificación de conceptos, y de la identificación de forma, definiendo como propiedades a obtener durante la identificación de la estructura: suficiencia, generalidad y aprendizaje.

Es de comentarse que este enfoque es aplicado en dominios orientados a tareas de logística para viajes aéreos y autobuses, lectura de mapas, simulación de vuelos autónomos, juntas y tutorías, quedando abierta la posibilidad de su aplicación en el área de toma de decisiones.

3.2 Generación de Argumentos con mayor credibilidad.

El trabajo de Hunter (Hunter, 2004) se discute sobre el impacto de los argumentos en términos de empatía y antipatía para definir una relación de preorden sobre los argumentos, con la finalidad de capturar cómo un argumento es más creíble que otro para una audiencia dada.

El trabajo de Hunter no utiliza un corpus para su desarrollo ni tampoco desarrolla un sistema de recomendación. Además de estar enfocado al problema de Selección, mientras que el trabajo de tesis realizado está dirigido al PPP.

3.3 Perspectiva Dialéctica en MCDA

En el trabajo de (Ouerdane, 2006) también aborda el desarrollo de un sistema de recomendación, donde se puede apreciar de forma abstracta una caracterización del diálogo a través de diferentes conjuntos de reglas, entre ellas de locución, de compromiso, y de seguimiento del diálogo.

Este trabajo aborda de forma genérica el problema de selección, donde se escoge un subconjunto a partir un universo de acciones, de acuerdo a preferencias de un decisor.

Derivado de la investigación presentada, al final del trabajo el autor comenta sobre la necesidad que existe de llevar a cabo un estudio que permita identificar criterios e indicadores que puedan ser útiles para clasificar las explicaciones, de acuerdo a los criterios de preferencia de los usuarios.

Se hace notar que, a pesar de que éste trabajo guarda una relación con la caracterización de diálogos en el área de toma de decisiones, aún no ha sido probado en un contexto particular como el que define el problema de cartera de proyectos, razón por la cual queda bajo estudio esta situación.

3.4 Framework de Apoyo a la Toma de decisiones para el PCP

En el trabajo de (Cruz-Reyes, 2014), se aborda la arquitectura de un sistema de recomendación para el PPP, donde a través de la teoría de argumentación y conjuntos rugosos, propone una caracterización de argumentos y diálogo.

Cabe mencionar que en este trabajo no utiliza ningún corpus, ni tampoco desarrolla un prototipo de sistema de recomendación, solo se queda en el modelo conceptual, lo cual si se desarrolla en este trabajo de investigación.

3.5 Análisis Comparativo

En resumen, se puede concluir a través de la lectura del estado del arte, que existen trabajos de caracterización del diálogo en diferentes áreas, sin embargo, no existe alguno de ellos que aborde problemas como el PPP.

Cabe mencionar que el uso de corpus, y el desarrollo de un prototipo que automatice el proceso de caracterización de la postura del DM, son contribuciones que el trabajo provee a la comunidad científica.

En la Tabla 2 se muestra un resumen de las características de cada trabajo presentado en el Estado del Arte. Se puede observar que la principal aportación es que se realizará el prototipo de un sistema de recomendación y la creación de un corpus de diálogo argumentativo.

Tabla 2. Resumen de las características sobre los trabajos del Estado del Arte.

Trabajo	Caracterización de argumentos	Arquitectura	Corpus	Problema	Sistema de Recomendación
Chotimongkol	Si	No	Uso	Orientado a Tareas	No
Hunter	Si	No	No	Selección	No
Ouerdane	Si	Si	No	Selección	Conceptual
Cruz-Reyes	Si	Si	No	Cartera de Proyectos	Conceptual

Este trabajo	Si	No	Uso y creación	Cartera de Proyectos	Conceptual y prototipo
--------------	----	----	----------------	----------------------	------------------------

Se puede observar a partir de la revisión de la literatura resumida en la Tabla 2 que los trabajos relacionados principalmente carecen de un prototipo funcional.

Además de que dichos trabajos están orientados a la argumentación dentro de un sistema de recomendación, y no a la automatización del proceso de interacción en el sistema de recomendación y el DM. Ésta última característica es lograda en este trabajo de investigación, a través de la estructura basada en el corpus de diálogo argumentativo.

Este trabajo de investigación tiene como ventajas que presenta un corpus que sirve de base de conocimiento para caracterizar a un DM y la construcción de un prototipo que de forma automática interactúa con un sistema de recomendación.

Capítulo 4. Propuesta de Solución

Como se ha mencionado en los apartados anteriores, se trabajó en la creación de un corpus de diálogo argumentativo que caracteriza la postura de un DM.

En vista de la literatura revisada con anterioridad y el debido análisis del Estado del Arte, se ha llegado a la conclusión que el sistema es factible de realizar, y se procedió al siguiente paso, que fue el diseño de una propuesta de solución para la caracterización de un DM.

Para ello, en esta sección se describen los diferentes elementos que conforman el sistema, su función y el lugar que les corresponde. Esto permitió sentar las bases para la implementación y posterior experimentación que dió resultados acerca de la eficiencia del prototipo.

4.1. Arquitectura General para la Caracterización de un DM

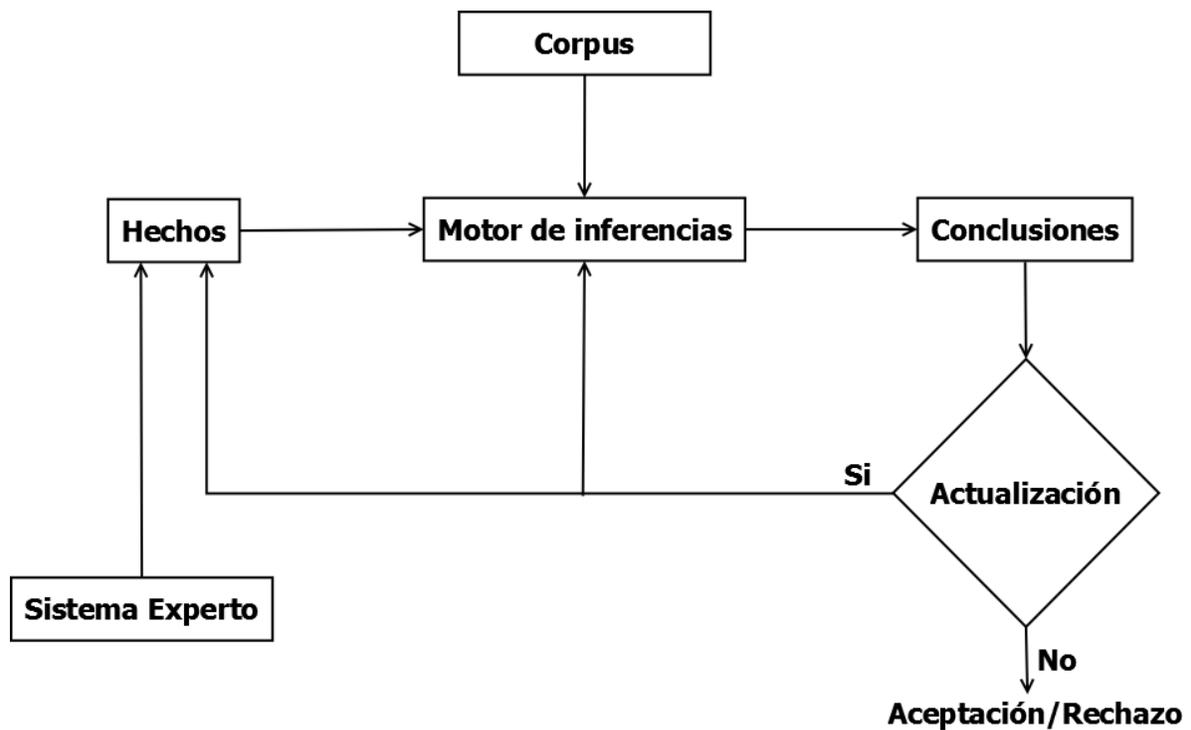


Figura 4.1. Diagrama General del procedimiento de Caracterización del Decisor y Diálogo.

La Figura 4.1, nos presenta los elementos que integran la propuesta de solución. Puede observarse que hay 6 módulos principales, los cuales interactúan iterativamente en un juego de diálogo, hasta llegar a una aceptación o rechazo de una recomendación (este punto se tratará más adelante en secciones posteriores).

Básicamente, la propuesta está compuesta de dos elementos principales:

1.- El DM. Este es emulado a través de la caracterización (se compone del corpus, hechos, motor de inferencias, conclusiones y actualización).

2.- El Sistema Experto. Es un sistema de recomendación, desarrollado para el PPP (visto en el capítulo 2), el cual permite dar una recomendación para ser aceptada o rechazada por el DM. Para esta propuesta, el sistema experto fue tomado de Macías (Macías, 2016).

A continuación, se detallan con mayor extensión cada uno de los elementos que conforman esta propuesta.

4.2 DM

El DM es el elemento principal en el trabajo de investigación. Por ello, la estrategia se enfocará en emular con la mayor fidelidad posible las decisiones tomadas por él. Este procedimiento consta de 5 módulos principales, que sostienen la caracterización del DM planteado. Cada uno se detalla en este apartado.

4.2.1 Hechos

En el juego de diálogo entre el sistema experto y el decisor artificial, existen argumentos distintivos que permiten respaldar la postura en un determinado momento. En la interacción con el sistema de recomendación, podemos hallar datos o información que nos permiten modificar la interacción de una manera racional.

Dentro del ámbito de PPP (Sección 2), podemos ver elementos representativos, como son los presupuestos de las carteras, los valores de los objetivos, las preferencias del DM, el estándar de prueba utilizado, etcétera.

Por ello, podemos definir un hecho como todo aquel argumento, información o dato comprobable, que tiene una relevancia tanto en la caracterización del DM, como en la interacción y actualización.

Los hechos permitirán guiar la caracterización del DM, permitiéndole actualizarse según las situaciones que vayan apareciendo dentro del diálogo. Un hecho está conformado por una categoría y su valor (por ejemplo máximo valor en objetivo 1 y con valor de 2).

4.2.2. Corpus

Para poder crear un DM artificial de una manera adecuada, es necesario contar con la mayor cantidad de información posible, pues esto permitirá emular con mayor fidelidad su postura y por ende, la toma de decisiones. Para definir el corpus se utilizó, se utilizó una base de datos que permite almacenar diferentes textos argumentativos que caracterizarán la postura del decisor artificial.

Dichos textos contendrán información relevante del DM. En el Anexo A, se presenta un ejemplo donde puede observarse con mayor detenimiento este proceso de caracterización. La caracterización son textos que representan al decisor, que luego son transformados a una estructura de árbol de argumentos para su utilización.

Los textos buscan representar elementos del DM en base al PPP, como pueden ser las preferencias hacia ciertos objetivos, un límite de presupuesto, una tolerancia hacia la realización de ciertas acciones para luego actualizar y cambiar de acciones, etcétera .

Para guardar dichos textos, se crea una base de datos que utiliza una estructura basada en el modelo de Toulmin (Ver sección 2.5.2), con la excepción de que solo se vieron involucrados 3 tipos de oraciones: *claim*, *premise* y *objection*.

Un *claim* es una declaración o argumento en forma de una oración afirmativa. Por ejemplo: La Luna está hecha de queso verde.

Mientras tanto, se le llama *premise* a cualquier frase que respalde a un enunciado *claim*. Por ejemplo: Las rocas de la Luna estaban verdes y olían a queso.

Por último, un *objection* es cualquier frase que provea evidencia contra un enunciado de tipo *claim*. Por ejemplo: Pero la mayoría de las rocas estaban hechas de granito.

Estas oraciones son colocadas en un árbol de argumentos dentro de la base de datos (cabe aclarar que este proceso no es automático sino manual). Además, cada frase posee un peso argumentativo, dado por el analista, que será de utilidad en los momentos donde se utilice el método de selección de conclusiones (ver Sección 5).

4.2.3. Motor de inferencias

Los árboles de argumentos poseen sentencias afirmativas o conclusiones (*claims*) que serán respaldadas por premisas y por oraciones negativas (*objections*).

Cabe mencionar, que tanto las premisas como las oraciones de tipo *objection*, poseen un valor o peso argumentativo que permitirán seleccionar de entre diversas conclusiones según la suma de sus pesos argumentativos. Por consiguiente, los enunciados de tipo *claim*, tienen la suma y resta de los valores argumentativos que se relacionan con ellos.

Entonces, dichas frases argumentativas son transformadas para generar reglas proposicionales que permitan interactuar con los hechos obtenidos por el Sistema Experto. El motor de inferencias es el encargado de esta transformación, ya que es el módulo encargado de transformar los diferentes árboles de argumentos en reglas proposicionales.

Para ello, se usó del lenguaje de programación prolog (Bratko, 1986). Estas serán de vital importancia para interactuar con los hechos extraídos del sistema experto. Estas reglas proposicionales, al colocarse en estados de verdadero o falso, contribuyen a un manejo sencillo de la información generada en el diálogo.

La finalidad de las reglas de prolog es el obtener conclusiones, con el cual el DM artificial toma la decisión de realizar una determinada acción.

4.2.4. Conclusiones

Las reglas de prolog (explicadas en el punto anterior) generarán conclusiones (debido a la salida de respuestas lógicas de falso y verdadero). Entonces, con base a dichas conclusiones, podremos pasar a diferentes situaciones.

Si al revisar las reglas proposicionales con los hechos, no genera ninguna conclusión, entonces estaremos ante un caso de cuestionar la recomendación. La explicación de esta acción, es que al no generar conclusiones, la recomendación no presenta coherencia con los datos proporcionados por el DM (por ejemplo, las preferencias).

En dado caso que si existan conclusiones, pueden suceder dos acciones distintas: si es una sola conclusión, entonces se emite directamente. En caso contrario, se procede a realizar el método estadístico, el cual se detalla en el siguiente punto.

4.2.5. Actualización

El proceso de actualización sucederá en dos situaciones diferentes. Una de ellas es cuando el decisor artificial haya emitido una conclusión. Si la conclusión es el retar al sistema experto para argumentar sobre los hechos que respaldan la recomendación, entonces el sistema generará más hechos al respecto, para poder sustentar la recomendación emitida.

En dado caso que la conclusión sea diferente, entonces, el Sistema podrá lanzar preguntas críticas hacia el decisor artificial, con el fin de que ahora sea él quien argumente los hechos a favor de su elección.

Ahora, recordemos que parte importante de la toma de decisiones, es que nunca se podrá eliminar el factor humano al momento de realizar este proceso. El diálogo puede estancarse en un ciclo, ya que el decisor artificial cuestiona, mientras que el sistema experto emite hechos.

Por ello, se ha tomado como estrategia para evitar estas situaciones, el caracterizar una tolerancia al número de iteraciones entre el decisor artificial y el sistema experto. Dicha tolerancia está expresado por medio de un árbol de argumentos, que posteriormente es transformado en una regla de prolog.

Si esta tolerancia llega a rebasarse, el DM genera un cambio en su postura, el cual se ve reflejado en la caracterización, permitiendo la salida dentro del ciclo del diálogo argumentativo.

Estos elementos involucrados en el diálogo, permitie la interacción entre ellos. Cuando el sistema experto y el DM artificial no tienen objeciones mutuamente, entonces se procederá a realizar la parte final del proceso: la aceptación o rechazo de la recomendación.

Como fue mencionado con anterioridad, el factor humano sigue presente, por lo que al final, queda en manos del decisor. Si no existen objeciones, se procede a aceptar dicha recomendación. En caso contrario, se rechazará.

4.3. Sistema Experto

Es el prototipo de Sistema de Recomendación que se utilizó para validar el funcionamiento del trabajo de tesis. Tiene diferentes funcionalidades, como encargarse de generar la recomendación inicial, además de brindar argumentos que respalden la recomendación emitida o cuestionar al decisor.

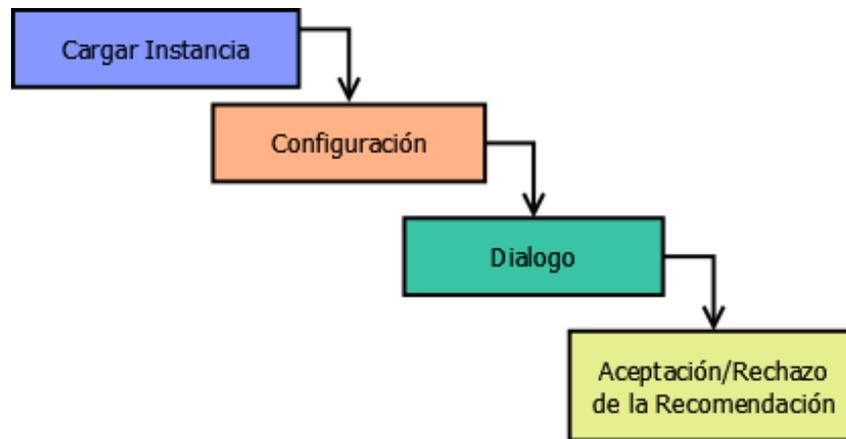


Figura 4.2. Diagrama Modular del Sistema Experto.

El sistema de recomendación planteado (Figura 4.2), tiene 4 etapas fundamentales, las cuales son cargar instancia, configuración, diálogo y la aceptación/rechazo de la recomendación (Macías, 2016). Cada una de las diferentes etapas, consta de diferentes subprocesos, los cuales se detallarán en el Capítulo 5.

Capítulo 5. Implementación del Prototipo

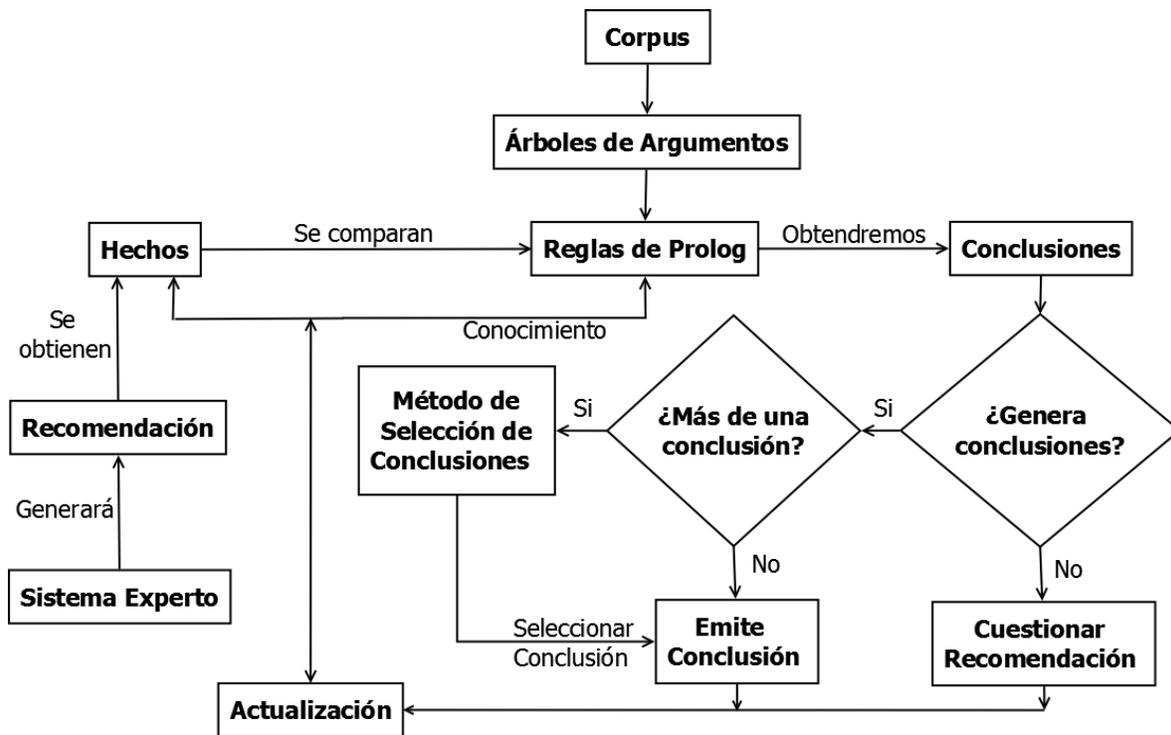


Figura 5.1. Procedimiento de Caracterización del Decisor y diálogo con el Sistema Experto.

Podemos observar con mayor precisión los elementos que componen el proceso de caracterización (Figura 5.1). Esto nos ayudará a entender con una exactitud mejor, la implementación del prototipo.

En forma general, se explicaron cada módulo que compone al proceso de caracterización (capítulo 4), y ahora se detallará la implementación. En el Anexo A, podrá observarse un ejemplo de caracterización más detallado, el cual facilitará el entendimiento del proceso.

Este capítulo describirá cada uno de los elementos mostrados en el diagrama de procedimiento de caracterización del DM (Figura 5.1). Los tres elementos principales son DM, sistema experto y el diagrama de clases de la implementación.

5.1. DM

5.1.1. Corpus

El corpus fue implementado como una base de datos en el motor MySQL, realizándose por medio de 7 tablas, donde vemos elementos principales (textos, decisores, etcétera).

Este abarca las 3 tablas enmarcadas en el recuadro rojo que vemos en la Figura 5.2 (Diagrama Entidad-Relación) y procedimientos almacenados que permiten agregar, borrar, actualizar y mostrar.

La tabla decisores permite almacenar DMs. La tabla textos permite almacenar lo textos argumentativos que caracterizan al DM, mientras que la tabla decisortexto enlaza a las dos tablas anteriores.

Las tablas *claim*, *premise* y *objection*, se asocian con los enunciados establecidos por Toulmin (ver sección 4.2.2), de tal modo que a cada frase se le asociará un peso argumentativo y un identificador para poder identificarlo. Por último, existe una tabla que permite unir los textos argumentativos con sus respectivos *claims*.

En el anexo A (Sección A.1), podemos ver un ejemplo de texto argumentativo que se va identificando sus elementos y transformando en información útil para el sistema. Los textos argumentativos son los que serán añadidos a la base de datos (que es el corpus). Este conjunto de información nos permitirá tener un DM artificial (Sección A.3).

5.1.2. Árboles de Argumentos

En la misma Figura 5.2, podemos observar cómo se encuentran resaltados en un recuadro verde 4 tablas (*claim*, *premise*, *objection* y *textoclaim*). Estas conforman la estructura de árbol de argumentos, basado en Toulmin.

Los textos argumentativos, se descomponen en los elementos de esta estructura (*claim*, *premise* y *objection*), asignándoles su peso correspondiente, para así poder servir al procedimiento de selección (ver Sección A.2 del Anexo A).

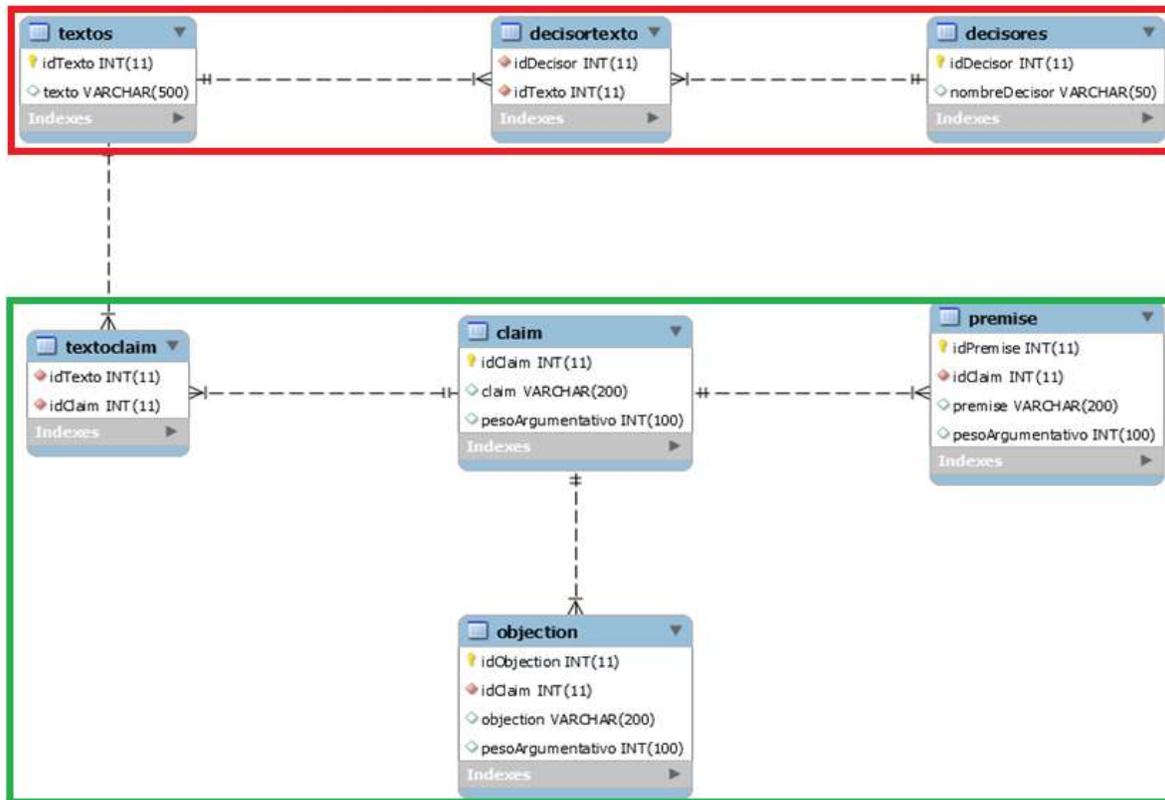


Figura 5.2 Diagrama Entidad-Relación de la base de datos del corpus.

5.1.3. Reglas de Prolog

Se mencionó en la sección 4.2.3 (motor de inferencia) que el corpus se transformaría en reglas proposicionales, por medio del lenguaje prolog. En la Figura 5.3, podemos observar la forma en que genera el archivo de prolog, con base en las reglas (*claim = premises + objections*) y los hechos que son extraídos durante el proceso.

El archivo de prolog está conformado primeramente por las reglas proposicionales. Después de esto, serán seguidas de los hechos (mejor y peor alternativa en cierto objetivo, prueba estándar, características del diálogo, etcétera). Podemos ver esto en la sección A.4.

Los primeros enunciados, son las diferentes reglas de prolog, seguidas de los diferentes hechos que se extrajeron.

Al generarse el diálogo entre el sistema experto y el DM artificial, es necesario presentar argumentos concretos por los cuales se está emitiendo una recomendación.

Por ello, además de la extracción inicial de hechos al presentar la recomendación inicial, a lo largo del diálogo se extraerán más hechos, con la finalidad de actualizar las reglas de Prolog y así permitir una mejor fidelidad al emular la postura del DM (Sección A.9).

```

bestalternative(X):-preferenceobjective1(X),preferenceobjective3(X).
preferenceobjective1(X):-bestobjective1(X),isveryexpensive(X).
preferenceobjective3(X):-bestobjective3(X),thelimitsofregionareclosetothelimit(X).
tolerance(3).
bestobjective1(3).
bestobjective2(1).
bestobjective3(3).
bestobjective4(2).
worstobjective1(2).
worstobjective2(3).
worstobjective3(1).
worstobjective4(3).
bestCriteria1(1).
bestCriteria2(1).
bestCriteria3(2).
bestCriteria4(2).
isveryexpensive(3).
isverycheap(1).
thelimitsofregionareclosetothelimit(3).
proofStandar(simplemajority).
ordinality(yes).
anonymity(yes).
additivityCoalitions(yes).
additivityValues(no).
veto(no).
farnessFromTheWorstSolution(no).

```

Figura 5.3 Script de salida de Prolog.

5.1.4. Conclusiones

Como se mencionó en el punto que trata los árboles de argumentos, cada conclusión se encuentra respaldada por premisas y refutada por objeciones.

Entonces, al encontrarnos con varias conclusiones, es necesario un mecanismo que nos permita escoger la acción más adecuada en el diálogo. Por ello, se decidió el utilizar la media aritmética (por su facilidad de implementación).

La media aritmética permitirá sumar el valor argumentativo positivo (premisas) y restar el negativo (objeciones), para luego dividir entre el total de elementos que permiten concluir los enunciados afirmativos (claim).

Posteriormente, se procederá a seleccionar la conclusión que tenga un mayor valor argumentativo. Si llegara a darse la situación en la cual hubiera un empate en el valor arrojado por la media aritmética como el más alto, entonces se procederá a elegir aleatoriamente entre dichas conclusiones, simulando así el proceso de razonamiento de un DM.

Esto puede verse más claramente en la sección A.7 del Anexo A.

5.1.5. Actualización

El proceso de actualización sucede en dos situaciones diferentes. Una de ellas es cuando el DM artificial haya emitido una conclusión. Si la conclusión es el retar al sistema experto para argumentar sobre los hechos que respaldan la recomendación, entonces el Sistema generará más hechos al respecto, para poder sustentar la recomendación emitida.

En dado caso que la conclusión sea diferente, entonces, el sistema puede lanzar preguntas críticas hacia el DM artificial, con el fin de que ahora sea él quien argumente los hechos a favor de su elección.

Ahora, recordemos que parte importante de la toma de decisiones, es que nunca se podrá eliminar el factor humano al momento de realizar este proceso. El diálogo puede estancarse en un ciclo, ya que el DM artificial cuestiona, mientras que el sistema experto emite hechos.

Por ello, se ha tomado como estrategia para evitar estas situaciones, el caracterizar una tolerancia al número de iteraciones entre el DM artificial y el sistema experto. Dicha tolerancia está expresado por medio de un árbol de argumentos, que posteriormente es transformado en una regla de prolog.

Si esta tolerancia llega a rebasarse, el DM generará un cambio en su postura, el cual se verá reflejado en la caracterización, permitiendo la salida dentro del ciclo del diálogo argumentativo. Esto se ve reflejado en la Sección A.8

Estos elementos involucrados en el diálogo, permitirán la interacción entre ellos. Cuando el sistema experto y el DM artificial no tienen objeciones mutuamente, entonces se procederá a realizar la parte final del proceso: la aceptación o rechazo de la recomendación.

Como fue mencionado con anterioridad, el factor humano seguirá presente, por lo que al final, quedará en manos del decisor. Si no existen objeciones, se procederá a aceptar dicha recomendación. En caso contrario, se rechazará.

5.2. Sistema Experto

El sistema experto es quien da la recomendación inicial (Sección A.5). En la sección 4.3, se mencionó que el sistema experto está compuesto por 4 módulos, estos son:

5.2.1. Módulo Cargar Instancia

Este módulo se encarga de recibir los valores definidos en la instancia seleccionada y almacenarlos en el sistema.

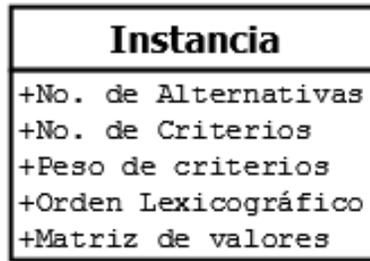


Figura 5.4. Clase Instancia.

La Figura 5.4 es el diagrama de clase que representa a la clase *instancia* dentro del sistema experto. Pueden observarse sus elementos clave. Cabe mencionar, que las instancias del sistema experto son diferentes a las del DM artificial (Ver Sección 6.2 para ver instancias de este trabajo de tesis).

5.2.2. Módulo Configuración

El proceso de configuración está compuesto por los siguientes módulos:

- Generar Premisas Iniciales
- Definir Reglas de Locución
- Generar Diagrama de Transición de Estados
- Definir Estándar de Prueba

Generar Premisas Iniciales

A partir de los datos de la instancia, el sistema genera un conjunto de premisas, que serán utilizadas en el juego de diálogo.

Dichas premisas sirven para definir la conclusión de cada uno de los argumentos utilizados en el diálogo con el objetivo de llegar a un acuerdo entre sistema y el usuario, ya sea de aceptación o rechazo (el procedimiento se muestra en la Figura 5.5).

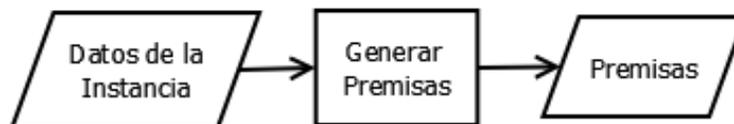


Figura 5.5. Diagrama de flujo Premisas iniciales.

Definir Reglas de Locución

Las reglas de locución definen los movimientos que un jugador puede realizar en el diálogo. Según Prakken (Prakken, 2005) cada regla de locución tiene como propiedades los movimientos, sus posibles contestación, la retracción y el estado de la declaración (un ejemplo lo vemos en la Figura 5.6).

Movimiento	Contestación	Retracción	Estado
afirmar(ϕ)	retar(ϕ) ¿Cuál prueba? formular(C, tipo, ϕ , id_esquema)	aceptar(ϕ)	Declarada

Figura 5.6. Ejemplo de Reglas de Locución

Generar Diagrama de Transición de Estados

En base a las reglas de locución formuladas, se genera un diagrama de transición de estados que limite los posibles movimientos disponibles tanto del usuario como el sistema en un respectivo estado (como el de la Figura 5.7), dicho diagrama está compuesto de:

- Estados
- Acciones de cada estado

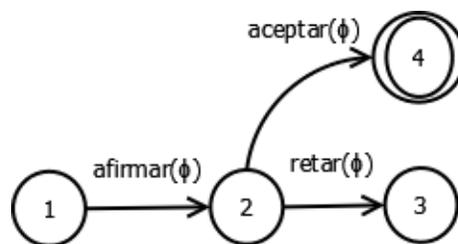


Figura 5.7. Ejemplo de Diagrama de Transición de Estados

Definir Estándar de Prueba

Apoyándose de los datos recibidos de la instancia, el sistema buscaseleccionar el Estándar de Prueba que más convenga utilizar, para posteriormente ser capaz de recomendar una solución que satisfaga al usuario o se acerque lo suficiente a sus preferencias (Figura 5.8).

Basándose en el trabajo de Ouerdane (Ouerdane, 2006), las propiedades que se utilizaron para definir el estándar de prueba a utilizar son: ordinalidad, anonimato, adición con peso, adición con valores, veto y dependencia entre criterios

De igual manera, los estándares de prueba que se consideraran son: mayoría simple con y sin veto, orden lexicográfico, mayoría de peso con y sin veto, suma de pesos y modelo no aditivo.

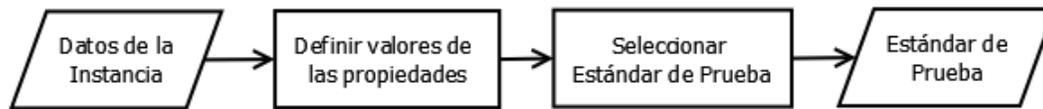


Figura 5.8. Diagrama de flujo Estándar de Prueba.

5.2.3. Módulo Diálogo

El proceso de diálogo está compuesto por los siguientes módulos:

- Recomendar Solución
 - Interacción
 - Corpus
- Estándar de Prueba
- Actualizar Criterios
- Actualizar Estándar de Prueba

Recomendar Solución

Utilizando el estándar de prueba seleccionado, el sistema recomienda la solución que ofrezca la mejor calidad. A lo largo del diálogo, el principal motivo de cuestionamiento será la recomendación.

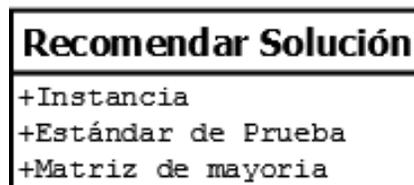


Figura 5.9. Clase Recomendar Solución.

La Figura 5.9 muestra los elementos que intervienen, para la selección de la recomendación inicial. El Estándar de prueba tiene una influencia directa en dicha recomendación, durante el juego de diálogo. Es por ello, que al cambiar de Estándar de prueba, es muy probable que haya cambio en la recomendación.

Interacción

En éste módulo se llevará a cabo el juego de diálogo entre el usuario y el sistema.

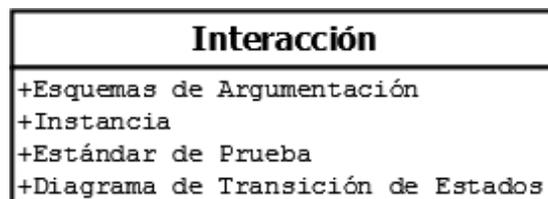


Figura 5.10. Clase Interacción.

La Figura 5.10 nos muestra los diferentes elementos que componen la interacción. A su vez, el corpus de diálogo argumentativo, es el DM artificial que interviene con el sistema experto.

Esquemas de Argumentación

Dentro del módulo de Interacción se hará uso de los esquemas de argumentación para definir la aceptación o rechazo de cada declaración establecida, ya sea para modificar propiedades de los estándares de prueba o para la aceptación o rechazo de la recomendación actual.

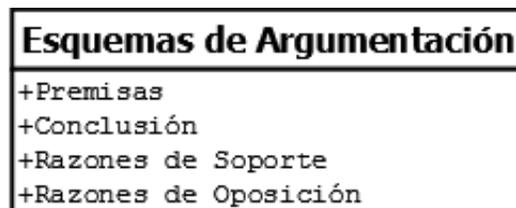


Figura 5.11. Clase Esquemas de Argumentación.

La Figura 5.11 nos permite observar los atributos que poseen los esquemas de argumentación. Dichos esquemas, modificarán las elecciones del sistema experto.

Módulos para cambio de propiedades

Una situación muy común, es el cambio de valores de la instancia. Por ello, el cambio de propiedades es fundamental. Podemos dividir en dos subcategorías:

- **Actualizar Criterios:** Puede ser utilizado en cualquier momento de la recomendación o interacción, genera una nueva recomendación a partir de la nueva información.
- **Actualizar Estándar de Prueba:** A partir de la información obtenida durante el proceso de interacción, el sistema puede modificar las propiedades de los estándares de prueba, por lo que debe volverse a generar una nueva recomendación utilizando el estándar de prueba más recomendable en base a los nuevos conocimientos obtenidos.

5.2.4. Módulo Aceptación/Rechazo de la Recomendación

Después del proceso de interacción, el usuario puede concluir si la recomendación obtenida es satisfactoria a su parecer o si ésta no lo convence aun cuando sea la mejor solución disponible.

Se establece una opción de rechazo debido a que el aspecto humano que estará en contacto con el sistema, al no seguir un conjunto rígido de reglas puede salirse del contexto y rechazar directamente una recomendación en vez de entrar en el juego de diálogo con el sistema.

5.3. Diagrama de Clases

Fueron utilizadas 5 clases. 4 de ellas pertenecen al presente trabajo (*Corpus*, *Claim*, *Objection* y *Premise*) y la 5ª es el Sistema de Recomendación en el cual se probó el corpus.

El corpus fue implementado con base en la metodología de solución. Se generaron textos argumentativos en la base de datos (la tabla de textos en el diagrama entidad-relación mostrada en la tabla anterior permite representar esta parte, donde dicho conjunto es el corpus).

Después de esto, los textos guardados en la base de datos, fueron transformados en árboles de argumentos con base en la estructura simplificada de Toulmin. El resultado es la conversión de textos argumentativos en 3 diferentes tipos de oraciones (explicados con anterioridad).

Cabe aclarar que este procedimiento no se encuentra automatizado, por lo cual es trabajo del analista la realización de dicha acción, incluyendo el dar el peso argumentativo a cada enunciado en dicho proceso.

Ahora, la conversión de los textos en árboles de argumentos, fue realizado en el método inicial, donde luego de realizar una consulta a la base de datos para extraer los enunciados *claim*, *objection* y *premise*, se colocaron en objetos de sus clases respectivas (podemos ver esto mejor en la Figura 5.2).

Esta estructura, junto con la extracción de hechos (detallada en el siguiente párrafo) permitirán la generación de las reglas de prolog (están ordenadas por un enunciado *claim*, seguido de todas sus premisas y objeciones).

A continuación, el método Hechos, procedió a extraer los hechos pertinentes del sistema de recomendación (este genera un conjunto de soluciones, para después recomendar una alternativa).

La información obtenida de los hechos, estará en el mismo lugar que las reglas de prolog, para poder interactuar de manera adecuada.

Cabe aclarar que este proceso de extracción de hechos se realizará continuamente, durante las diferentes iteraciones del diálogo, ya que se presentarán situaciones determinadas por el diagrama de transición de estados, donde la actualización de hechos será necesaria (por ejemplo, un cambio en la prueba estándar o en las propiedades de diálogo, etcétera).

Dichos cambios son manejados por un método de actualización, el cual actuará para retroalimentar y generar nuevo conocimiento que permita proseguir la interacción.

Es necesario mencionar que la tolerancia juega un papel fundamental, ya que esto permitirá la realización preguntas críticas, y dicha acción generará cambios mayores en el juego de diálogo entre el sistema de recomendación y el corpus realizado en este trabajo.

Los hechos extraídos permiten generar conclusiones con base al archivo de reglas de prolog y generar las conclusiones que cumplan con los hechos extraídos durante el proceso de interacción.

Estas conclusiones permitirán el seleccionar acciones dentro del diagrama de transición de estados, tales como argumentar, retar, realizar preguntas críticas, etcétera (todo esto está anclado dentro de un ciclo que procederá mientras no se llegue a un estado final de aceptación o rechazo).

Como se mencionó con anterioridad, puede suceder que existan diferentes casos de conclusiones. Por ello, el número de conclusiones (representado por el método Conclusiones), permite la interacción entre los hechos y las reglas de prolog.

Como se habló en la metodología de solución, la cantidad de soluciones tendrá una repercusión directa en la interacción del diálogo (Ninguna conclusión, una sola conclusión o varias).

Cuando surgen más de una conclusión, el método de selección de conclusiones es el de la media aritmética (sumar pesos argumentativos y verificar el que tenga un mayor valor positivo).

Para finalizar esta sección, podemos ver en la Figura 5.3 como están relacionadas las clases en la implementación. Tanto el sistema experto como el corpus de diálogo, interactúan entre sí, permitiendo un juego de diálogo que desencadenará las acciones presentadas a lo largo de este capítulo.

Este es una de las aportaciones que genera este trabajo de tesis. Podemos observar la estructura utilizada para la caracterización de un DM implementada y como interactúa con el sistema de recomendación. También podemos ver la estructura de Toolmin simplificada, a través de las 3 clases que interactúan con el corpus (*objection*, *premise* y *claim*).

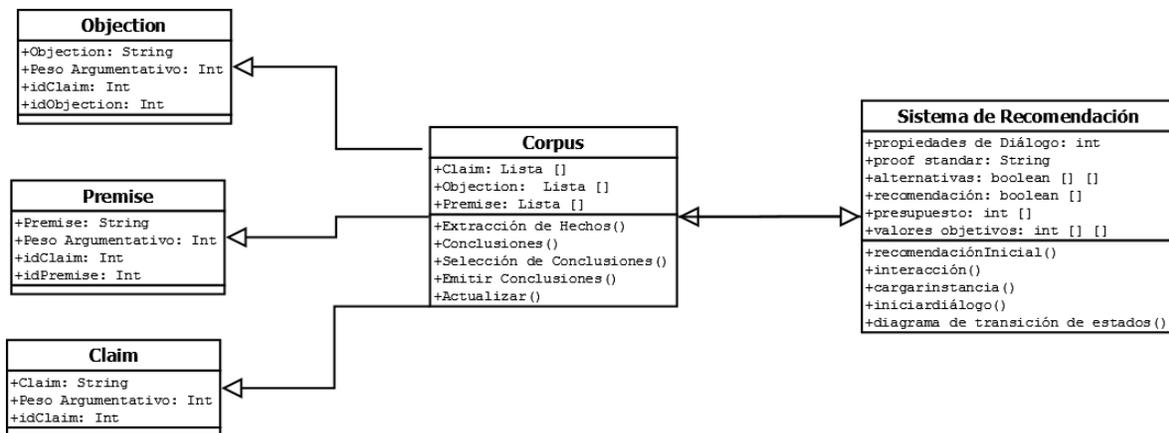


Figura 5.12. Diagrama de Clases.

Capítulo 6. Experimentación y Resultados

En este capítulo, se presentarán los resultados de la experimentación. Esta consistió en la caracterización del DM por medio de diálogos argumentativos, que permiten emular su postura. Para ello, se realizaron interacciones entre un individuo y el sistema de recomendación de Macías (Macías 2016). Con base a los diálogos obtenidos se generaron textos argumentativos, los cuales permitieron la creación de un DM artificial, el cual interactuaría con el sistema de recomendación de Macías, para así evaluar su rendimiento al comprobar la fidelidad del diálogo simulado con respecto al original. Se obtuvieron 30 diálogos distintos de un individuo con el sistema de recomendación. Tras haber caracterizado los diálogos, se ejecutó, obteniendo los resultados que se mostrarán en la sección 6.2.

Se mostrarán primeramente las condiciones de experimentación (Hardware y software, el sistema de recomendación, la caracterización del Usuario e instancias), para después presentar los resultados obtenidos, realizando una prueba estadística para verificar el rendimiento.

6.1. Condiciones de Experimentación

La experimentación fue realizada en un equipo de cómputo con las siguientes características:

Hardware

- Procesador Intel Core i5 de 1.6 GHz.
- GB de memoria RAM.
- Sistema Operativo Windows 8 de 64 bits.

Software

- Netbeans IDE 8.02 con el compilador Java JDK7. Fue utilizado para realizar la conexión entre el corpus de diálogo argumentativo, prolog y el sistema de recomendación. Además, este lenguaje de programación es donde fue implementada la mayor parte del código.
- SWI-PROLOG 7.2.3 de 64 bits. Este es el motor de inferencias. Se utilizó para interactuar con las reglas de prolog que se utilizaron durante la implementación.

- MySQL Workbench 6.3. Este fue el motor de base de datos donde se implementó el corpus de diálogo argumentativo y la estructura de árbol de argumentos.

6.1.1. Sistema de Recomendación

El sistema de recomendación utilizado para la validación, es el reportado por Macías (Macías, 2016, en proceso).

Las acciones del usuario están caracterizadas según el trabajo de Ouerdane (Ouerdane, 2006), se puede ver más claramente en el diagrama de transición de estados que se presentará a continuación en la Figura 6.1:

- Retar (Uchallenge). Solicita que un enunciado pueda servir como base para justificar o explicar.
- Aceptar (Uaccept). Permite aceptar un *claim*.
- Pregunta Crítica (UposeC). Son preguntas que se utilizan para retar un argumento, utilizando una pregunta asociada con una postura de locución.
- Aseverar (Uassert). Permite intercambiar información.
- Argumentar (Uargue). Son las respuestas para sustentar a un *claim*.
- Rechazar (Uretract). Esta locución es para el rechazo de un *claim*.

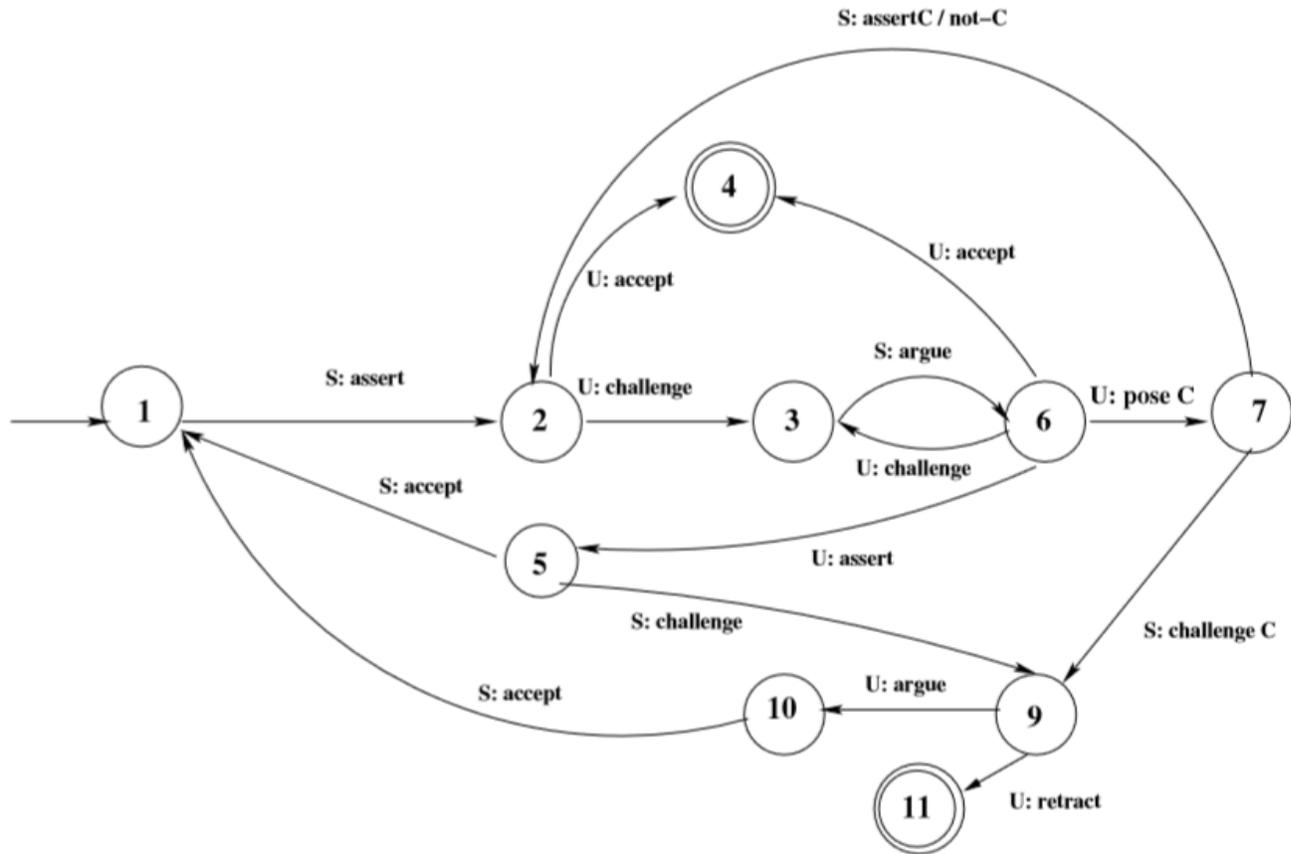


Figura 6.1. Diagrama de Transición de Estados del Sistema de Recomendación.

El diagrama de transición de estados nos permite ver como las diferentes acciones tienen un orden de aparición, el cual permite acotar que estén disponibles ciertas acciones a lo largo de cada iteración en el juego de diálogo entre el sistema de recomendación y el DM.

6.1.2. Caracterización del Usuario en base al Sistema de Recomendación

El sistema de recomendación presenta las siguientes opciones de respuesta. Los valores de respuesta ya están predefinidos dentro del sistema, y no se pueden cambiar. Es importante mencionar que la experimentación quedó acotada a estas respuestas (ver Tabla 3), donde un argumento se puede rechazar (REC) o aceptar (ACE).

Tabla 3. Opciones de respuesta del Sistema de Recomendación.

Retar	<ul style="list-style-type: none"> ➤ Why? (RET1) ➤ Why is this alternative better than other solutions? (RET2)
Pregunta Critica	<ul style="list-style-type: none"> ➤ Is the current proof standard the best option? (PC1) ➤ Will the recommended solution still be the same after the proof standard is changed? (PC2) ➤ Does the system have enough information to define a proof standard? (PC3) ➤ Does the system have a preference for a certain criterion? (PC4) ➤ Are you an honest source? (PC5) ➤ Are the reasons that support the recommendation strong enough? (PC6) ➤ How much better is the recommendation in comparison with the rest of the solutions? (PC7) ➤ Did you assert that the recommendation is true? (PC8) ➤ Are you an honest source? (PC9) ➤ How satisfactory is the recommendation? (PC10) ➤ How far has the dialogue progressed? (PC11) ➤ Are you in a position to know if the recommendation is true? (PC12) ➤ Is the difference between the support reasons and the opposition big enough? (PC13)
Aseverar	<ul style="list-style-type: none"> ➤ I like my choice more than the recommendation (ASE1) ➤ The discrepancy between the preferred criteria values on the recommended portfolio and another solution isn't significant (ASE2) ➤ The budget for this portfolio is too close to the maximum limit, I would like to choose another one (ASE3) ➤ I have a preference for a certain criterion (ASE4)
Argumentar	<ul style="list-style-type: none"> ➤ I only care about the cardinal information in the performance. Ordinality activated (AR1) ➤ I care about all the information available. Ordinality deactivated (AR2) ➤ All criteria are exchangeable. Anonymity activated (AR3) ➤ All criteria cannot be exchanged. Anonymity deactivated (AR4) ➤ A coalition of criteria is relevant. Additivity w.r.t. coalitions activated (AR5) ➤ A coalition of criteria is irrelevant. Additivity w.r.t. coalitions deactivated (AR6) ➤ I want a sum of the values of each criterion. Additivity w.r.t. values activated. (AR7) ➤ I don't want a sum of the values of each criterion. Additivity w.r.t. values deactivated (AR8) ➤ I want to use a veto threshold. Veto activated (AR9) ➤ I don't want to use a veto threshold. Veto deactivated (AR10) ➤ I want my solution to be as far as possible from the worst solution. Farness from the worst solution activated. (AR11) ➤ I only care about the best solution. Farness from the worst solution deactivated (AR12) ➤ Simple majority (AR13) ➤ Lexicographic method (AR14) ➤ Weighted majority (AR15) ➤ Weighted sum (AR16) ➤ TOPSIS (AR17) ➤ Keep using the current one (AR18) ➤ Simple majority + veto (AR19) ➤ Weighted majority + veto (AR20) ➤ Preferencia Criterio (AR21) ➤ No Changes (AR22)

Además de estas acciones, el DM puede actualizar los valores en cualquier momento (esta acción estará denominada como ACT en las secciones siguientes).

6.1.3 Instancias

Las instancias de diálogo del sistema de recomendación, fueron organizadas como pueden verse en la Tabla 4. Originalmente, los diálogos se encontraban visibles como se muestran en la Figura 6.2 (se utilizó un prototipo de sistema de recomendación para ello), como parte del área de diálogo.

Para cada instancia, solo se tomó el área del usuario y sus acciones, ya que estas son las que se busca caracterizar en el diálogo. Una vez que se tuvieron las instancias, se ejecutó para cada instancia, de tal modo que se obtuvieran los diálogos simulados.

Después, ambos diálogos (original y simulado) se compararon con el número de acciones realizadas por cada una y el orden de secuencia, para ver la fidelidad del diálogo y las discrepancias entre ellos.

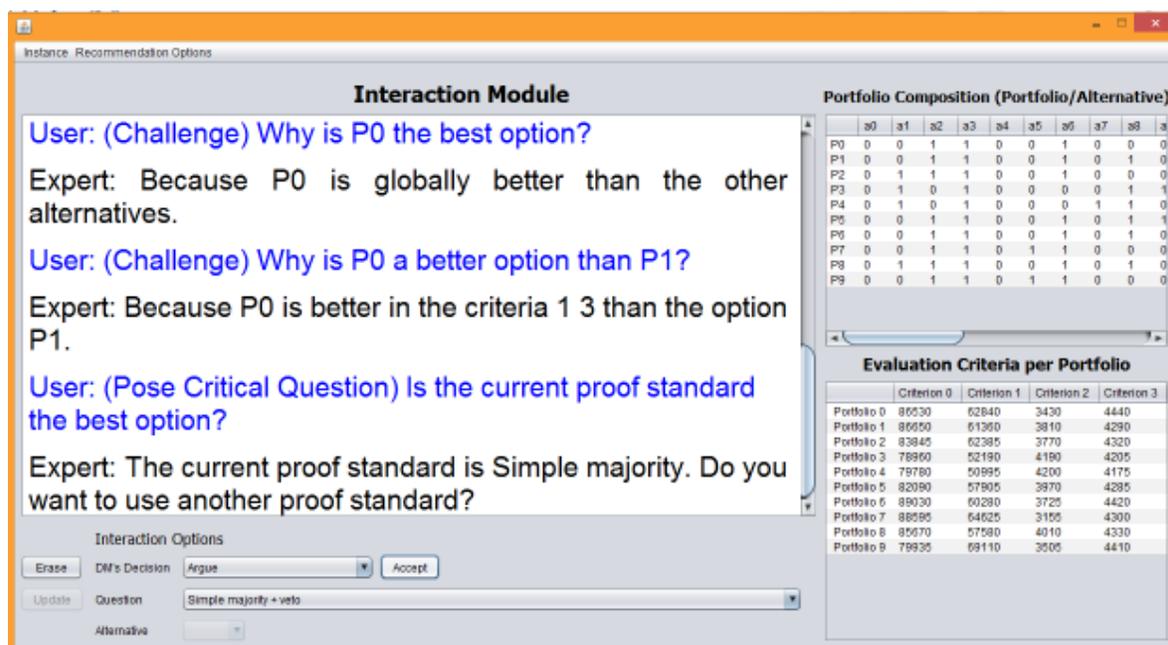


Figura 6.2. Ejemplo de Diálogo del Sistema de Recomendación.

A continuación, se muestran los diálogos de las 30 instancias. Cada instancia, presenta las diferentes acciones que realizaba el DM durante la interacción con el sistema experto (las acciones posibles fueron establecidas en el punto 6.1.4).

Dichas acciones permiten que cada diálogo sea diferente y pueda ser modelado en la caracterización del usuario y poder alcanzar el objetivo el tener una mayor fidelidad a la hora de reproducir el diálogo.

Tabla 4 Instancias de la Experimentación

Instancia	Diálogo Original
1	RET1 (P0), PC3, AR9, RET2 (P6, P2), ACE.
2	RET1 (P0), RET2 (P0, P8), PC1, AR15, RET1 (P9), ASE1 (P6), AR21 (C0), ACE.
3	RET2 (P0, P6), PC3, AR21 (C0), RET (P6), PC4, AR22, ACE.
4	RET1 (P0), PC1, AR14, RET2 (P6, P0), ACE.
5	RET2 (P0, P6), ASE1 (P6), AR21 (C0), RET (P6, P0), PC1, AR17, RET2 (P9, P6), ASE1 (P6), ASE1, ACE.
6	RET1 (P0), PC2, RET2 (P0, P6), PC1, AR20, RET1 (P9), RET2 (P9, P6), PC5, RET1 (P9), ASE1, AR21 (C0), ACE.
7	RET2 (P0, P6), PC6, RET1 (P0), PC7, RET2 (P0, P1), PC3, AR12, RET1 (P0), ASE1, ASE3, RET2 (P0, P6), ASE1, AR21 (C0), ACE.
8	RET1 (P0), PC8, RET2 (P0, P6), PC4, AR21 (C0), RET1 (P6), PC2, RET1 (P6), RET2 (P6, P7), PC1, AR19, RET1 (P6), ACE.
9	RET1 (P0), RET2 (P0, P1), PC1, AR22, RET2 (P0, P6), PC4, AR21 (C0), RET1 (P6), ACE.
10	ACT (Alternativa 21: 3350), RET1 (P0), PC4, AR21 (C0), RET1 (P7), ACE.
11	ACT (Alternativa 21: 3350), RET2 (P0, P7), ASE1 (P7), AR21 (C0), ACE.
12	ACT (Alternativa 21: 3350), RET1 (P0), PC1, AR19, RET1 (P0), PC2, RET2 (P0, P6), PC1, AR14, ACE.
13	ACT (Alternativa 21: 3350), RET1 (P0), ASE1 (P7), ASE1, ACE.
14	RET2 (P0, P6), ACT (Alternativa 21: 3350), RET2 (P0, P7), PC3, AR22, RET1 (P0), PC4, AR21 (C0), ACE.
15	RET2 (P0, P6), PC2, RET1 (P0), PC4, AR21 (C0), ACT (Alternativa 21: 3350), RET1 (P7), ACE.
16	RET2 (P0, P6), RET2 (P0, P7), PC6, RET1 (P0), PC9, RET1 (P0), PC4, AR21 (C0), ACT (Alternativa 21: 3350), ACE.
17	RET1 (P0), ASE1 (P6), ASE1, ACT (Alternativa 21: 3350), RET1 (P0), ASE1 (P7), AR21 (C0), ACE.
18	RET1 (P0), RET2 (P0, P7), ACT (Alternativa 21: 3350), RET2 (P0, P7), ASE1 (P1), ASE1 ACE.

Instancia	Diálogo Original
19	RET2 (P0, P6), PC3, AR3, ACE.
20	RET2 (P0, P6), PC3, AR4, ACT (Alternativa 21: 3350), ACE.
21	RET1 (P0), PC10, RET2 (P0, P6), PC7, RET2 (P0, P7), ASE1 (P6), AR21 (C0), ACE.
22	RET2 (P0, P6), PC1, AR20, RET2 (P9, P6), ACT (Alternativa 21: 3350), RET1 (P9), PC11, RET2 (P9, P7), PC1, AR17, RET1 (P9), ASE1 (P7), AR21 (C0), ACE.
23	ACT (Alternativa 13:8675), RET1 (P0), ASE1 (P1), AR21 (C0), ACE.
24	ACT (Alternativa 13:8675), RET2 (P0, P1), PC1, AR14, ACE.
25	ACT (Alternativa 13:8675), ACT (Alternativa 21: 3350), RET2 (P0, P1?), ASE1 (P1), ASE1, RET1 (P0), ASE1 (P1), AR21 (C0), ACE.
26	RET1 (P0), PC5, ACT (Alternativa 13:8675), RET2 (P0, P1), PC2, ACT (Alternativa 21: 3350), RET2 (P0, P1), PC3, AR4, RET1 (P1), ACE.
27	RET1 (P0), PC1, AR15, RET2 (P9, P6), ACT (Alternativa 13:8675), RET2 (P9, P1?), ASE1 (P1), AR21 (C0), ACE.
28	RET1 (P0), PC7, RET2 (P0, P6), ACT (Alternativa 13:8675), RET1 (P0), PC4, AR21 (C0), ACE.
29	RET2 (P0, P1), ASE1 (P1), AR21 (C0), ACT (Alternativa 13:8675), ACE.
30	RET1 (P0), PC12), RET2 (P0, P6), PC13, ACT (Alternativa 13:8675), RET2 (P0, P1), PC4, AR4, ACE.

Con base al sistema de recomendación explicado en la Sección 6.1.4. Se utilizaron las siguientes abreviaturas, con la finalidad de que la lectura de las tablas de resultados que se presentan sea más sencilla y entendible:

- RET representa retar.
- ACE representa aceptar.
- PC representa pregunta crítica.
- AS representa aseverar.
- AR representa argumentar.
- REC representa rechazar.
- ACT representa actualizar.

Cabe recordar que dichas acciones, son las presentadas en el diagrama de transición de estados en la Figura 6.1.

Una vez planteadas las abreviaturas que se utilizaron, se procede contabilizar el número de ocurrencias por instancia, para cada acción disponible. La información es mostrada en la Tabla 5.

Tabla 5. Información obtenida de las instancias

Instancia	RET	ACE	PC	AS	AR	REC	ACT
1	2	1	1	0	1	0	0
2	3	1	1	1	2	0	0
3	2	1	2	0	2	0	0
4	2	1	1	0	1	0	0
5	3	1	1	2	3	0	0
6	5	1	3	1	2	0	0
7	5	1	3	2	3	0	0
8	7	1	4	0	2	0	0
9	4	1	2	0	2	0	0
10	2	1	1	0	1	0	1
11	1	1	0	1	1	0	1
12	3	1	3	0	2	0	1
13	1	1	0	1	1	0	1
14	3	1	2	0	2	0	1
15	3	1	2	0	1	0	1
16	4	1	3	0	1	0	1
17	2	1	0	2	2	0	1
18	3	1	0	1	1	0	1
19	1	1	1	0	1	0	0
20	1	1	1	0	1	0	1
21	3	1	2	1	1	0	0
22	5	1	3	1	3	0	1
23	1	1	0	1	1	0	1
24	1	1	1	0	1	0	1
25	2	1	0	2	2	0	2
26	4	1	3	0	1	0	2
27	3	1	1	1	2	0	1
28	3	1	2	0	1	0	1
29	1	1	0	1	1	0	1
30	3	1	3	0	1	0	1

Ahora que tenemos un resultado cuantitativo de la frecuencia de cada acción, podremos realizar un análisis sobre el desempeño de los diálogos simulados por el DM artificial.

6.2 Resultados

En la Tabla 6, se muestran los diálogos generados por el DM artificial, cuya implementación ya fue explicada a detalle en el Capítulo 5. La estructura de la presentación es idéntica a la Tabla 4, sin embargo, puede notarse como hay diferencias entre los diálogos originales y los simulados.

Tabla 6. Diálogos Emulados

Instancia	Diálogo Emulado
1	RET1 (P0), PC3, AR9, RET1 (P6), RET2 (P6, P2), ACE.
2	RET1 (P0), RET2 (P0, P6), PC1 AR14, RET1 (P6), RET2 (P0,P6), ACE.
3	RET2 (P0, P6), PC3, AR21 (C0), RET1 (P6), RET2 (P0, P6), PC4, AR21 (C0), ACE.
4	RET1 (P0), PC1, AR22, RET1 (P0), ASE1 (P6), AR21 (C0), ACE.
5	RET2 (P0, P6), ASE1 (P6), AR21 (C0), RET2 (P6, P0), PC1, AR22, RET1 (P6), ACE.
6	RET1 (P0), PC2, RET2 (P0, P6), PC1, AR9, RET1 (P9), RET2 (P9, P6), ASE1 (P6), ACE.
7	RET2 (P0, P6), PC2, RET1 (P0), PC7, RET2 (P0, P1), PC3, AR22, RET1 (P0), RET2 (P0, P6), ASE1 (P6), ASE3, ACE.
8	RET1 (P0), PC8, RET2 (P0, P6), PC4, AR21 (C0), RET1 (P6), PC2, ACE.
9	RET1 (P0), RET2 (P0, P6), PC1, AR14, RET1 (P6), ACE.
10	ACT (Alternativa 21: 3350), RET1 (P0), PC4, AR21 (C0), RET1 (P7), RET2 (P7, P0) ACE.
11	RET2 (P0, P6), ASE1 (P6)), AR21 (C0), ACE.
12	ACT (Alternativa 21: 3350), RET1 (P0), PC1, AR19, RET1 (P0), PC1, AR14, ACE.
13	ACT (Alternativa 21: 3350), RET1 (P0), RET2 (P0, P7), ASE1 (P7), ASE1, ACE.
14	RET2 (P0, P6), RET2 (P0, P7), PC3, AR22, RET1 (P0), PC4, AR21 (C0), ACE.
15	RET2 (P0, P6), PC2, RET1 (P0), PC4, AR21 (C0), RET1 (P6), RET2 (P6, P0), ACE.
16	RET2 (P0, P6), PC6, RET1 (P0), PC9, RET1 (P0), PC4, AR21 (C0), ACE.
17	RET1 (P0), ASE1 (P6), ASE1, ACT (Alternativa 21: 3350), RET1 (P0), RET2 (P0, P7), ASE1 (P7), AR21 (C0), RET1 (P7), RET2 (P7, P0), ACE.
18	RET1 (P0), ACT (Alternativa 21: 3350), RET1 (P0), RET2 (P0, P7), ASE1 (P1), ASE1, ACE.
19	RET1 (P0), RET2 (P0, P6), PC3, AR4, ACE.
20	RET2 (P0, P6?), PC3, AR4, ACT (Alternativa 21: 3350), RET2 (P0, P6), ACE.
21	RET1 (P0), PC10, RET2 (P0, P6?), PC4, AR22, RET1 (P0), RET2 (P0, P6), ASE1 (P6), ASE1, ACE.
22	RET2 (P0, P6), PC1, AR20, RET2 (P9, P6), ACT (Alternativa 21: 3350), RET1 (P9), RET2 (P9, P7), PC1, AR22, RET1 (P9), ASE1 (P7), AR21 (C0), ACE.
23	ACT (Alternativa 13:8675), RET1 (P0), ASE1 (P1), AR21 (C0), RET1 (P0), ACE.
24	ACT (Alternativa 13:8675), RET1 (P0), RET2 (P0, P1), PC1, AR14, ACE.
25	ACT (Alternativa 13:8675), ACT (Alternativa 21: 3350), RET2 (P0, P1), ASE1 (P1), ASE1, ACE.

Instancia	Diálogo Emulado
26	RET1 (P0), PC5, ACT (Alternativa 13:8675), RET2 (P0, P1), PC2, RET2 (P0, P1), PC3, AR4, ACE.
27	RET1 (P0), PC1, AR15, RET2 (P9, P6), RET2 (P0, P1?), ASE1 (P6), AR21 (C0), ACE.
28	RET1 (P0), PC7, RET2 (P0, P6), RET1 (P0), PC4, AR21 (C0), ACE.
29	RET2 (P0, P1), ASE1 (P1), ACT (Alternativa 13:8675), AR21 (C0), ACE.
30	RET1 (P0), PC12, RET2 (P0, P6), RET1 (P0), PC13, ACT (Alternativa 13:8675), ACE.

Al igual que se realizó una contabilización de las ocurrencias de cada acción en cada instancia de los diálogos originales, la Tabla 7 presenta la información cuantitativa de ocurrencias de cada acción. Solo que en esta ocasión, fueron generadas por un DM artificial.

Tabla 7. Información obtenida de diálogos simulados

Instancia	RET	ACE	PC	AS	AR	REC	ACT
1	3	1	1	0	1	0	0
2	4	1	1	0	1	0	0
3	3	1	2	0	2	0	0
4	2	1	1	2	2	0	0
5	3	1	1	1	2	0	0
6	5	1	3	1	2	0	0
7	5	1	3	1	2	0	0
8	4	1	4	0	1	0	0
9	3	1	1	0	1	0	0
10	3	1	1	0	1	0	1
11	1	1	0	1	1	0	0
12	2	1	2	0	2	0	1
13	2	1	0	1	1	0	1
14	3	1	2	0	2	0	0
15	4	1	2	0	1	0	0
16	3	1	3	0	1	0	0
17	5	1	0	2	2	0	1
18	4	1	0	1	1	0	1
19	2	1	1	0	1	0	0
20	2	1	1	0	1	0	1
21	4	1	2	1	2	0	0
22	6	1	3	1	3	0	1
23	2	1	0	1	1	0	1
24	2	1	1	0	1	0	1
25	1	1	0	1	1	0	2
26	3	1	3	0	1	0	1
27	3	1	1	1	2	0	0
28	3	1	2	0	1	0	0
29	1	1	0	1	1	0	1
30	2	1	2	0	0	0	1

Con base a las Tablas 5 y 7, podremos establecer una diferencia cuantitativa entre los diálogos originales y los simulados. Esto se realiza al calcular el valor absoluto de la diferencia entre el número de acciones en ambas instancias. Por ejemplo $|2-3| = |-1| = 1$.

Por ello, la Tabla 8 nos permite ver esa información de manera detallada.

Tabla 8. Comparación de Diálogos y discrepancias entre ellos.

Instancia	RET	ACE	PC	AS	AR	REC	ACT	Total Discrepancias
1	1	0	0	0	0	0	0	1
2	1	0	0	1	1	0	0	3
3	1	0	0	0	0	0	0	1
4	0	0	0	2	1	0	0	3
5	0	0	0	1	1	0	0	2
6	0	0	0	0	1	0	0	1
7	0	0	0	1	1	0	0	2
8	3	0	0	0	1	0	0	4
9	1	0	1	0	1	0	0	3
10	1	0	0	0	0	0	0	1
11	0	0	0	0	0	0	1	1
12	1	0	1	0	0	0	0	2
13	1	0	0	0	0	0	1	1
14	0	0	0	0	0	0	1	1
15	1	0	0	0	0	0	1	2
16	1	0	0	0	0	0	1	2
17	3	0	0	0	0	0	1	3
18	1	0	0	0	0	0	0	1
19	1	0	0	0	0	0	0	1
20	1	0	0	0	0	0	0	1
21	1	0	0	0	1	0	0	2
22	1	0	0	0	0	0	0	1
23	1	0	0	0	0	0	0	1
24	1	0	0	0	0	0	0	1
25	1	0	0	1	1	0	0	3
26	1	0	0	0	0	0	1	2
27	0	0	0	0	0	0	1	1
28	0	0	0	0	0	0	1	1
29	0	0	0	0	0	0	1	1
30	1	0	1	0	1	0	1	3

Finalmente, para verificar la exactitud que tiene el DM en reproducir el diálogo original, se presenta la Tabla 9. En ella podemos ver las secuencias en las instancias (originales y simuladas) y podemos obtener datos importantes de su comportamiento. A continuación, se presentan los resultados de la comparación de secuencia de diálogo:

Tabla 9. Comparación de Orden de Aparición.

Instancia	Diálogo Original	Diálogo Simulado	Discrepancias
1	RET, PC, AR, RET, ACE.	RET, PC, AR, RET, RET ACE.	2
2	RET, RET, PC, AR, RET, AS, AR, ACE.	RET, RET, PC, AR, RET, RET, ACE.	3
3	RET, PC, AR, RET, PC, AR, ACE.	RET, PC, AR, RET, RET, PC, AR, ACE.	4
4	RET, PC, AR, RET, ACE.	RET, PC, AR, RET, AS, AR, ACE.	3
5	RET, AS, AR, RET, PC, AR, RET, AS, AR, ACE.	RET, AS, AR, RET, PC, AR, RET, ACE.	3
6	RET, PC, RET, PC, AR, RET, RET, PC, RET, ASE, AR, ACE.	RET, PC, RET, PC, AR, RET, RET, PC, RET, ASE, ACE.	2
7	RET, PC, RET, PC, RET, PC, AR, RET, AS, AR, RET, AS, AR, ACE.	RET, PC, RET, PC, RET, PC, AR, RET, RET, AS, AR, ACE.	6
8	RET, PC, RET, PC, AR, RET, PC, RET, RET, PC, AR, RET, ACE.	RET, PC, RET, PC, AR, RET, PC, RET, PC, ACE.	5
9	RET, RET, PC, AR, RET, PC, AR, RET, ACE.	RET, RET, PC, AR, RET, ACE.	3
10	ACT, RET, PC, AR, RET, ACE.	ACT, RET, PC, AR, RET, RET ACE.	2
11	ACT, RET, AS, AR, ACE.	RET, AS, AR, ACE.	4
12	ACT, RET, PC, AR, RET, PC, RET, PC, AR, ACE.	ACT, RET, PC, AR, RET, PC, AR, ACE.	4
13	ACT, RET, AS, AR, ACE.	ACT, RET, RET, AS, AR, ACE.	4
14	RET, ACT, RET, PC, AR, RET, PC, AR, ACE.	RET, RET, PC, AR, RET, PC, AR, ACE.	8
15	RET, PC, RET, PC, AR, ACT, RET, ACE.	RET, PC, RET, PC, AR, RET, RET, ACE.	3
16	RET, RET, PC, RET, PC, RET, PC, AR, ACT, ACE.	RET, PC, RET, PC, RET, PC, AR, ACE.	9
17	RET, AS, AR, ACT, RET, AS, AR, ACE.	RET, AS, AR, ACT, RET, RET, AS, AR, RET, RET, ACE.	6
18	RET, RET, ACT, RET, AS, AR, ACE.	RET, RET, ACT, RET, RET, AS, AR, ACE.	4
19	RET, PC, AR, ACE.	RET, RET, PC, AR, ACE.	4
20	RET, PC, AR, ACT, ACE.	RET, PC, AR, ACT, RET, ACE.	2

Instancia	Diálogo Original	Diálogo Simulado	Discrepancias
21	RET, PC, RET, PC, RET, AS, AR, ACE.	RET, PC, RET, PC, AR, RET, RET, AS, AR, ACE.	6
22	RET, PC, AR, RET ACT, RET, PC, RET, PC, AR, RET, AS, AR, ACE.	RET, PC, AR, RET, ACT, RET, RET, PC, RET, PC, AR, RET, AS, AR, ACE.	9
23	ACT, RET, AS, AR, ACE.	ACT, RET, AS, AR, RET, ACE.	2
24	ACT, RET, PC, AR, ACE.	ACT, RET, RET, PC, AR, ACE.	4
25	ACT, ACT, RET, AS, AR, RET, AS, AR, ACE.	ACT, ACT, RET, AS, AR, ACE.	3
26	RET, PC, ACT, RET, PC, ACT, RET, PC, AR, RET, ACE.	RET, PC, ACT, RET, PC, RET, PC, AR, ACE.	6
27	RET, PC, AR, RET, ACT, RET, AS, AR, ACE.	RET, PC, AR, RET, RET, AS, AR, ACE.	5
28	RET, PC, RET, ACT, RET, PC, AR, ACE.	RET, PC, RET, RET, PC, AR, ACE.	4
29	RET, AS, AR, ACT, ACE.	RET, AS, ACT, AR, ACE.	3
30	RET, PC, RET, PC, ACT, RET, PC, AR, ACE.	RET, PC, RET, PC, ACT, ACE.	3

Podemos notar como existen discrepancias entre los diálogos originales y los simulados. Por ello, deducimos de los resultados que es posible la caracterización de un DM. En el siguiente capítulo, se darán conclusiones en base a la información obtenida.

6.2.1. Prueba Estadística

Con la finalidad de verificar si existe una diferencia significativa entre las discrepancias obtenidas por los diálogos del prototipo y los diálogos reales, se utilizaron los siguientes datos.

Discrepancias en el orden de aparición

Máximo 9

Mínimo 2

Media = 4.2

Varianza = 3.82069

Desviación Estándar = 1.921805

Discrepancias en el contero de acciones

Máximo 4

Mínimo 1

Media = 1.7

Varianza = 0.906897

Desviación Estándar = 0.936305

Podemos notar que la diferencia en cuanto al orden de aparición es mucho mayor que la diferencia en el conteo de acciones (esto puede observarse en los valores de la varianza y la desviación estándar).

Se utilizó la página web tec.citius (Rodríguez-Fdez, 2015) para poder realizar una prueba no paramétrica de múltiples grupos. Podemos notar que la hipótesis nula es aceptada en cada una de las acciones (excepto en la opción retar), ya que no existen diferencias significativas entre ellos. Podemos observar los resultados en la Figura 6.3.

Comparison	Statistic	Adjusted p-value	Result
DM Real vs Instancia	8.01635	0.00000	H0 is rejected
DM Real vs Total Discrepancias	6.50262	0.00000	H0 is rejected
DM Real vs RET	3.88025	0.00094	H0 is rejected
DM Real vs ACT	1.98277	0.42654	H0 is accepted
DM Real vs AR	1.68429	0.82914	H0 is accepted
DM Real vs AS	0.87412	1.00000	H0 is accepted
DM Real vs PC	0.51168	1.00000	H0 is accepted
DM Real vs ACE	0.00000	1.00000	H0 is accepted
DM Real vs REC	0.00000	1.00000	H0 is accepted

Figura 6.3. Resultados de la prueba estadística.

Capítulo 7. Conclusiones y trabajos futuros

7.1 Conclusiones

Se logró alcanzar el objetivo perseguido por el presente trabajo de investigación, el cuál fue la caracterización del DM a través de un corpus de diálogo argumentativo.

Para poder llevar a cabo la caracterización fueron necesarios los siguientes elementos: 1) Textos Argumentativos; 2) Corpus, 3), Motor de Inferencia, 4) Validación de Conclusiones, 5) Actualización.

A través de la caracterización, también se logró automatizar el proceso de interacción entre un sistema de recomendación y un DM. Como resultado se obtuvo un prototipo al respecto. Dentro de los detalles a resaltar que fueron necesarios para la realización del prototipo están:

- La validación del corpus de diálogo argumentativo ha sido realizada con discrepancias entre el usuario y el generado por el sistema de recomendación. Con ello, se ha demostrado que es posible caracterizar un DM, utilizando la estrategia planteada a lo largo de este trabajo de tesis.
- Además, podemos observar como el corpus de diálogo argumentativo ya permite la interacción con el sistema de recomendación. Sin embargo, puede notarse que existe un factor de aleatoriedad, visto en la experimentación.
- Fue necesario implementar una estrategia de selección de conclusiones, para situaciones donde más de una era posible. En este trabajo de investigación, se escogió usar la media aritmética sobre el peso argumentativo (y criterio aleatorio para desempate final), debido a su sencillez de implementación y utilidad.

Finalmente, haciendo uso del modelo de caracterización del DM a través de corpus de diálogos argumentativos, y del prototipo, se pudo observar que mientras mejor sea la caracterización del DM, los resultados son más fieles al diálogo original.

Sin embargo, la aleatoriedad todavía sigue presente, es decir, siguen apareciendo parejas de conclusiones con la misma posibilidad de selección.

7.2 Aportaciones

Las aportaciones obtenidas a partir del trabajo de investigación, en relación con el PPP, se resumen a continuación:

1. Simular posturas del DM. Se logró caracterizar la postura de un DM a través de la combinación de textos argumentativos y el modelo Toulmin. Ambos constituyeron un corpus que se almacenó en una base de datos relacional.
2. Interacción con un sistema de recomendación. Se logró automatizar éste proceso, se diseñó un prototipo que haciendo el corpus generaba conclusiones válidas para un sistema de recomendación.

7.3 Trabajos futuros

La transformación automática de textos argumentativos a árboles de argumentos, no fue contemplada durante este proyecto, quedando como un trabajo a futuro. Este procedimiento es realizado por el analista.

Además, el peso argumentativo otorgado a cada uno de los argumentos, aun se realiza de forma manual, por medio del análisis que realiza el analista, quedando como trabajo a futuro la otorgación de dicho peso en forma automática.

En este trabajo se utilizó la media aritmética como estrategia para la selección de conclusiones, debido a su facilidad de implementación.

Por ello, la utilización de una estrategia diferente (como una prueba estadística) queda propuesta como un trabajo a futuro, para así tener una mejor selección de conclusiones y poder comparar el rendimiento entre las diferentes estrategias.

Anexo A.

Ejemplo de Caracterización de Usuario.

A.1 Texto Argumentativo

Texto argumentativo (BD->textos; BD->decisortexto; BD->decisores)

“Los proyectos caros pero con un gran impacto social los prefiero a los proyectos económicos.”

Los textos argumentativos serán parte fundamental de la caracterización del DM.

A.2 Árbol de Argumentos

El texto argumentativo, transformado a la estructura de Toulmin es:

Claim: A) *Prefiero proyectos*

Premises: B) *Proyectos con gran impacto social*

C) *Proyectos caros*

Objection: D) *Proyectos económicos*

La siguiente fase, consiste en la transformación de los textos argumentativos en los diferentes tipos de enunciados (*claim*, *premise* y *objection*). Esta acción es realizada por el analista.

Árbol de Argumentos (intermedio entre análisis y BDD):

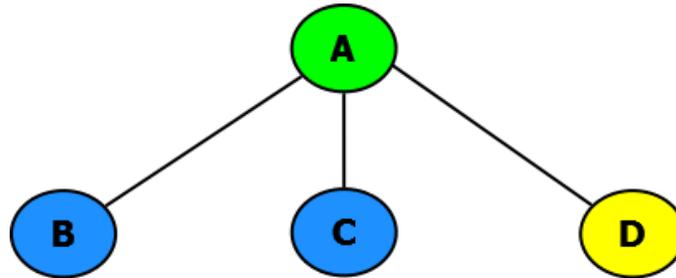


Figura A.1. Texto argumentativo, transformado en Árbol de Argumentos.

A.3 Inserción en Base de Datos

Base de Datos para el Árbol de Argumentos (BD->claim; BD->premise;
BD>objection;BD->textoclaim):

El siguiente paso, es la inserción de los diferentes enunciados en las tablas de la base de datos, según su tipo. Hay tres cosas importantes a tomar en cuenta:

1. El id es automático para identificar de forma única a cada enunciado (*claim premise y objection*).
2. Como se ha mencionado con anterioridad, el peso argumentativo es dado por el analista. Se utiliza en la prueba estadística, que de acuerdo al diagrama, se utilizará cuando hay más de una conclusión posible.
3. En el caso de las premisas y objeciones, estos están asociados a sus respectivos *claim*, por medio del id.

Siguiendo la estructura, tenemos lo siguiente:

- *Claim* (idClaim, claim, pesoArgumentativo)

A (1, Prefiero proyectos, 50)

- *Premise* (idPremise, idClaim, premise, pesoArgumentativo)

B (2, 1, Proyectos con gran impacto social, 60)

C (3, 1, Proyectos caros, 40)

- *Objection* (idObjection, idClaim, objection, pesoArgumentativo)

D (4, 1, Proyectos económicos, 40)

A.4 Reglas de Prolog

Una vez que cada enunciado es insertado en la base de datos, se procede a utilizar la interfaz de aplicación en Java, que extrae el árbol de argumentos y lo transforma en reglas de prolog, de la forma mostrada en la Figura A.2:

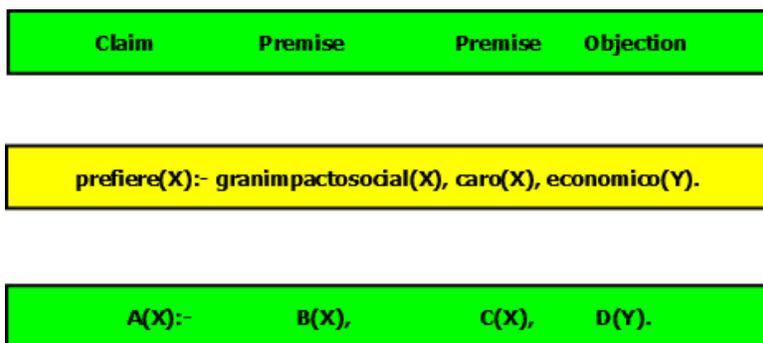


Figura A.2. Representación de reglas en Prolog.

Una vez que el sistema ha generado la lista de alternativas (Figura A.3) y da la recomendación inicial (Figura A.4), se procederá a la extracción de hechos por parte del corpus:

A.5 Recomendación Inicial

Cartera	h_0	h_1	h_2	h_3	h_4	Presupuesto	Área	Región
a_0	7	6	2	3	5	\$ 25,000	100	80
a_1	6	4	8	4	7	\$ 27,000	70	90
a_2	3	2	5	2	3	\$ 26,000	80	80
a_3	7	7	2	0	2	\$ 24,000	50	120

Figura A.3. Ejemplo de carteras válidas generadas por el Sistema Experto.

Cartera	h_0	h_1	h_2	h_3	h_4	Presupuesto	Área	Región
a_1	6	4	8	4	7	\$ 27,000	70	90

Figura A.4. Ejemplo de recomendación generada por el Sistema Experto.

A.6 Hechos

Al interactuar con el sistema de recomendación, se extraerán hechos con base a la respuesta del sistema de recomendación, como lo muestra la siguiente Figura A.5.

```

bestobjective1(3).
bestobjective2(1).
bestobjective3(3).
bestobjective4(2).
worstobjective1(2).
worstobjective2(3).
worstobjective3(1).
worstobjective4(3).
bestCriteria1(1).
bestCriteria2(1).
bestCriteria3(2).
bestCriteria4(2).
isveryexpensive(3).
isverycheap(1).
thelimitsofregionarecloseothelimit(3).
proofStandar(simplemajority).
ordinality(yes).
anonymity(yes).
additivityCoalitions(yes).
additivityValues(no).
veto(no).
farnessFromTheWorstSolution(no).

```

Figura A.5. Hechos obtenidos por parte del Sistema de Recomendación.

A.7 Conclusiones

Los hechos interactuarán con las reglas de prolog para generar conclusiones, para poder obtener los valores que corresponden (Figura A.6):

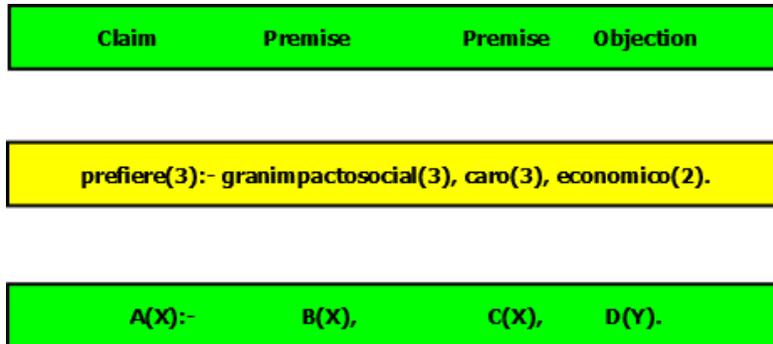


Figura A.6. Caracterización de reglas.

Esta acción se realiza para saber si las conclusiones son verdaderas o falsas, para así proceder al proceso de interacción, al seleccionar una acción según el diagrama de transición de estados.

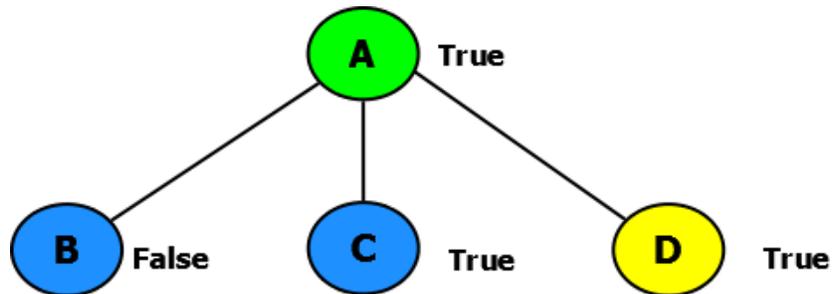


Figura A.7. Obtención de conclusiones.

En este ejemplo (Figura A.7), como A es verdadera (la preferencia del decisor), se procede a aceptar la solución (emitir la conclusión). En este caso específico, la alternativa es la adecuada.

A.8 Selección de Conclusiones

Sin embargo, supongamos que no hubiera más de una conclusión, se procedería a utilizar el método de selección de conclusiones.

Supongamos que en el ejemplo anterior, el enunciado A es falso (es decir, la alternativa no se encuentra dentro de las preferencias del DM), se pueden realizar acciones como retar (Challenge) o realizar una pregunta crítica (PoseC).

Primero que nada, caracterizaremos el Challenge (retar) y el PoseC (pregunta crítica) mostradas en la Figura A.8:

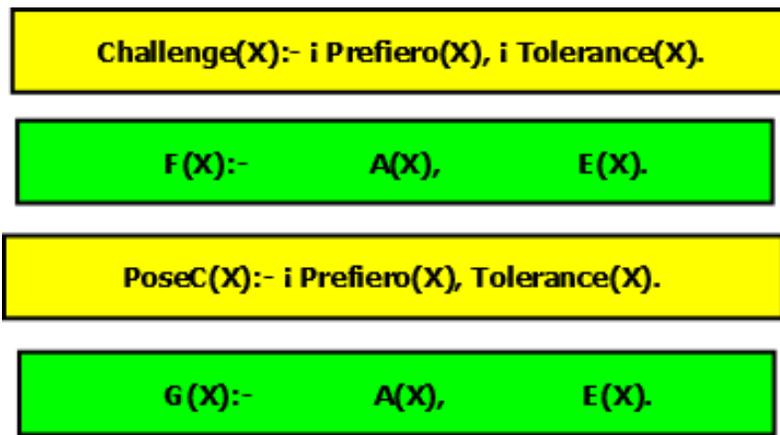


Figura A.8. Caracterización de las acciones Retar y pregunta crítica.

Entonces, al tener ambas conclusiones como verdaderas, se procederá a utilizar la media aritmética como método para seleccionar conclusiones y se decidirá por aquella que tenga el mayor valor argumentativo (en este caso, la pregunta crítica, según la **Figura A.9**):

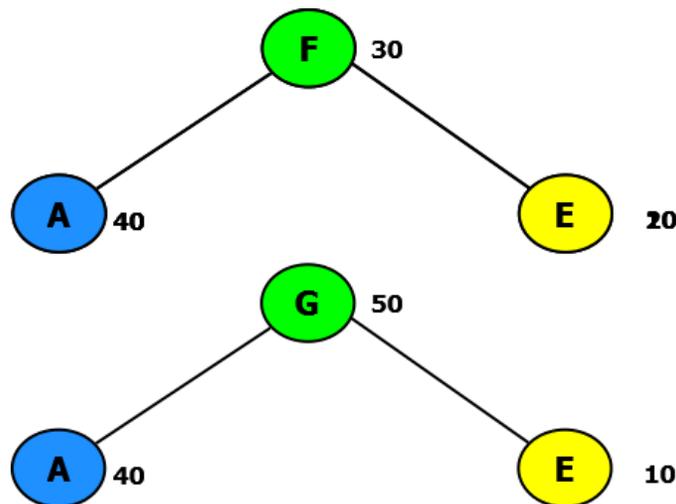


Figura A.9. Selección de conclusiones por medio de la media aritmética.

A.9 Actualización

Una vez emitida la conclusión (en este caso, la pregunta crítica o PoseC), se emite la conclusión como una acción en el Sistema de Recomendación, y por último, se actualiza el conocimiento en las reglas de prolog, según la acción emitida.

En el caso anterior, al ser una pregunta crítica, puede suceder que se pida cambiar la prueba estándar, las preferencias, propiedades de diálogo, etcétera, lo cual generará cambios en el conocimiento, haciendo que las reglas de prolog sean cambiadas, según sea el caso.

```
bestobjective1(3).
bestobjective2(1).
bestobjective3(3).
bestobjective4(2).
worstobjective1(2).
worstobjective2(3).
worstobjective3(1).
worstobjective4(3).
bestCriteria1(1).
bestCriteria2(1).
bestCriteria3(2).
bestCriteria4(2).
isveryexpensive(3).
isverycheap(1).
thelimitsofregionarecloseothelimit(3).
proofStandar(simplemajority).
ordinality(yes).
anonymity(yes).
additivityCoalitions(yes).
additivityValues(no).
veto(no).
farnessFromTheWorstSolution(no).
```

Figura A.10. Actualización de hechos.

Referencias Bibliográficas

Adra, S. F., Griffin, I., Fleming, P. J. (2007). A comparative study of progressive preference articulation techniques for multiobjective optimisation. In Proceedin

gs of the 4th international conference on Evolutionary Multi-criterion Optimization (EMO'07), Springer, Matsushima, Japan, pp. 908–921.

Amgoud L., Parsons S., Maudet N. (2000). Arguments, dialogue, and negotiation. In W. Horn, editor, Proceedings of the European Conference on Artificial Intelligence (ECAI'00), pages 338–342. IOS Press.

Amgoud L., Bonnefon, J.-F., Prade, H. (2005). An argumentation-based approach to multiple criteria decision. In L. Godo, editor, Proceedings of the 8th European Conference on Symbolic and Quantitative Approaches to Reasoning and Uncertainty (ECSQARU'05), pp. 269 – 280.

- Balderas F. (2011) Sistema de Apoyo a la Decisión para la Selección de Carteras de Proyectos en Organizaciones Públicas basado en Agentes. Tesis de maestría, Instituto Tecnológico de Cd Madero, Tamaulipas, México.
- Benzmüller, C., Schiller, M., & Siekmann, J. (2010). Resource-bounded modelling and analysis of human-level interactive proofs. In *Resource-adaptive cognitive processes* (pp. 291-311). Springer Berlin Heidelberg.
- Biran, O., & Rambow, O. (2011). Identifying justifications in written dialogs by classifying text as argumentative. *International Journal of Semantic Computing*, 5(04), 363-381.
- Botley, P. (2014). Three Very Different Translators: Joseph Scaliger, Isaac Casaubon and Richard Thomson. *Canadian Review of Comparative Literature/Revue Canadienne de Littérature Comparée*, 41(4),477-491.
- Bratko, I. (1986). *Prolog Programming for Artificial Intelligence*. Addison-Wesley, Reading, Massachusetts.
- Carazo, A. F., Gómez, T., Molina, J., Hernández-Díaz, A. G., Guerreo, F. M., Caballero, R. (2010). Solving a comprehensive model for multiobjective project portfolio selection. *Computers & Operations Research*, 37(4):630–639.
- Carlson, L. (1983). *Dialogue games*. Synthese Language Library. D. Reidel.
- Castro, M. (2007). Desarrollo e implementación de un framework para la formación de carteras de proyectos de I&D en organizaciones públicas. Tesis de Maestría, Universidad Autónoma de Nuevo León.
- Chotimongkol, A. (2008). Learning the structure of task-oriented conversations from the corpus of in-domain dialogs (Doctoral dissertation, SRI International).
- Cohon, J.L., Marks, D.H. (1975). A review and evaluation of multiobjective programming techniques. *Water Resources Research*, 11(2):208–220.
- Cruz-Reyes, L., Trejo, C. M., Irrarragorri, F. L., & Santillán, C. G. G. (2014). A Decision Support System Framework for Public Project Portfolio Selection with Argumentation Theory. In *Recent Advances on Hybrid Approaches for Designing Intelligent Systems* (pp. 467-479). Springer International Publishing.
- Deb, K. (2001). *Multi-objective optimization using evolutionary algorithms*. Interscience series in systems and optimization. John Wiley & Sons, LTD.
- Doerner, K., Gutjahr, W. J., Hartl, R., Strauss, C., Stummer, C. (2004). Pareto ant colony optimization: A metaheuristic approach to multiobjective portfolio selection. *Annals OR*, 131:79–99.
- Dung, P. M. (1995). On the Acceptability of Arguments and its Fundamental Role in Nonmonotonic Reasoning, Logic Programming and n-person games. *Artificial Intelligence*, 77(2):321–358.
- Edgeworth, F. (1881). *Mathematical Psychics*. P. Keagan, London, England.

- Fernández, E., López, E., Navarro, J., Vega, I. (2011): “Aplicación de metaheurísticas multiobjetivo a la solución de problemas de cartera de proyectos públicos con una valoración multidimensional de su impacto”, *Gestión y Política Pública*, vol. XX, no. 2, pp. 381-432.
- García, R. (2010). Hiper-heurístico para resolver el problema de cartera de proyectos sociales. Tesis de Maestría, Instituto Tecnológico de Cd. Madero.
- Gendreau, M., Potvin, J-Y. (2005). Meta-heuristics in combinatorial optimization, *Annals of Operations Research* 140 (1) 189-213.
- Goldberg, E. (1989) *Genetic Algorithms in Search, Optimization and Machine Learning*. AddisonWesley Publishing Company, Reading, Massachusetts.
- Hunter, A. (2004, July). Making argumentation more believable. In *AAAI* (Vol. 4, pp. 269-274).
- Keeney, R.L., Raiffa, H. (1993) *Decisions with Multiple Objectives: Preferences and Value Tradeoffs*. Cambridge University Press.
- Larousse, P. (2015). *Diccionario Larousse* Available at: <http://www.larousse.mx/>.
- Laumanns, M., Thiele, L., Deb, K., Zitzler, E. (2002) Combining convergence and diversity in evolutionary multi-objective optimization. *Evolutionary Computation*, 10(3):263 – 282.
- Li, K., Kwong, S., Cao, J., Li, M., Zheng, J., Shen, R. (2012). Achieving balance between proximity and diversity in multi-objective evolutionary algorithm, *Information Sciences* 182 (1): 220–242.
- Luke, S. (2009). *Essentials of Metaheuristics*. Available at <http://cs.gmu.edu/sean/book/metaheuristics/>.
- Macías, T (2016). Desarrollo de esquemas de argumentación para sistemas de recomendación en toma de decisiones. En proceso.
- McCarthy J.L. (1977). Epistemological problems of artificial intelligence. In R. Reddy, editor, *Proceedings of the 5th International Joint Conference on Artificial Intelligence (IJCAI'77)*, pages 1038–1044.
- McDermott, D., Doyle, J. (1980). Nonmonotonic logic 1. *Artificial Intelligence*, 13:41–72.
- Moens, M. F., Boiy, E., Palau, R. M., & Reed, C. (2007, June). Automatic detection of arguments in legal texts. In *Proceedings of the 11th international conference on Artificial intelligence and law* (pp. 225-230). ACM.
- Navarro, J. (2005). Herramientas inteligentes para la evaluación y selección de proyectos de investigación-desarrollo en el sector público. PhD thesis, Doctoral Thesis, Universidad Autónoma de Sinaloa, Sinaloa, Mexico.
- Norman, T.J., Carbogim, D.V, Krabb, E.C., Walton, D. (2003). Argument and Multi-Agent Systems. In C. Reed and T.J. Norman, editors, *Argumentation Machines: New Frontiers in Argument and Computation*, chapter 2, pp. 15–54. Kluwer Academic.

- Ouerdane, W. (2006), Multiple Criteria Decision Aiding: a Dialectical Perspective. PhD thesis, Université Paris Dauphine.
- Oztürk, M., Tsoukiàs, A., Vincke, Ph.. (2005). Preference Modelling, In: State of the Art in Multiple Criteria Decision Analysis, M. Ehrgott, Greco, S. and Figueira, J. (Ed.). Wiley Series on Intelligent Systems. Springer-Verlag. pp. 27 – 72.
- Pareto, V. (1896). Cours d'économie politique, volume 1 & 2. Lausanne.
- Parsons S., Sierra C., Jennings, N. (1998). Agents that reason and negotiate by arguing. *Journal of Logic and Computation*, 8(3):261–292
- Pérez, F. Molina, J., Caballero, R., Coello, C., Hernández, A., (2007). Hibridación de métodos exactos y heurísticos para el problema multiobjetivo. *Journal Economic Literature*: C61; C63. XV Jornadas de ASEPUMA y III Encuentro Internacional.
- Prakken,H. (2001). Relating protocols for dynamic dispute with logics for defeasible argumentation. *Synthese*, 127:187–219.
- Reed, C., & Rowe, G. (2004). Araucaria: Software for argument analysis, diagramming and representation. *International Journal on Artificial Intelligence Tools*, 13(04), 961-979.
- Reiter, R. (1980). A logic for default reasoning. *Artificial Intelligence*, 67(1, 2):81–132.
- Reidel, D. (1983). *Dialogue Games: an Approach to Discourse Analysis*. Synthese Language Library. Dordrecht.
- Reiter, P. (2010). *Metaheuristic Algorithms for Solving Multi-objective/Stochastic Scheduling and Routing Problems*. Tesis Doctoral, University of Wien.
- Rodríguez-Fdez, I., Canosa, A., Mucientes, M., Bugarín, A. (2015) STAC: a web platform for the comparison of algorithms using statistical tests, in: Proceedings of the 2015 IEEE International Conference on Fuzzy Systems (FUZZ-IEEE). Recuperado de <http://tec.citius.usc.es/stac/index.html>.
- Roy, B. (1990) Reading in Multiple Criteria Decision Aid, chapter The Outranking Approach and the Foundations of ELECTRE methods, pages 155–183. Springer-Verlag.
- Roy, B. (1996). *Multicriteria Methodology for Decision Aiding*. Kluwer Academic, Dordrecht.
- Sánchez, P. (2007). *Modelos para la combinación de preferencias en toma de decisiones: herramientas y aplicaciones*. Tesis Doctoral, Universidad de Granada.
- Santana, L. (2004). *Un Algoritmo Basado en Evolución Diferencial para Resolver Problemas Multiobjetivo*. Tesis de Maestría, Centro de Investigación y de Estudios avanzados del Instituto Politécnico Nacional, Departamento de Ingeniería Eléctrica sección de Computación.
- Serrano, V. (2007). *Métodos para reducir evaluaciones en algoritmos evolutivos multi-objetivo, basados en aproximación de funciones*. Tesis de Maestría, Centro de Investigación

y de Estudios avanzados del Instituto Politécnico Nacional, Departamento de Ingeniería Eléctrica sección de Computación.

Tan, K., Goh, C., Mamun, A., Ei, E. (2008). An evolutionary artificial immune system for multiobjective optimization. *European Journal of Operational Research*, 187(2):371 – 392.

Toulmin, S. (1958). *The Uses of Argument* Cambridge University Press. Cambridge, England.

Tsoukias, A. (2011). Aiding to Decide. Concepts and Issues. In *Evaluation and Decision Models, Real Case Studies*. Springer, Berlin.

Van Veldhuizen, D., Lamont, G. (1998). Evolutionary Computation and Convergence to a Pareto Front. In: *Late Breaking Papers at the Genetic Programming Conference, University of Wisconsin, Madison, Wisconsin, USA, Stanford University Bookstore*. John R. Koza, J.R. (Ed.). Morgan Kaufmann pp. 221-228.

Verheij, B. (2003). Dialectical argumentation with argumentation schemes: an approach to legal logic. *Artificial Intelligence and Law*, 11(2-3):167–195.

Walton, D. (1996). *Argumentation schemes for Presumptive Reasoning*. Mahwah, N. J., Erlbaum.

Walton, D., Reed, C. (2002). Argumentation schemes and defeasible inferences. In G. Carenini, F. Grasso, and C. Reed, editors, *Workshop on Computational Models of Natural Argument (CMNA '02)*.

Walton, D.N. (2005). Justification of argument schemes. *Australasian journal of logic*, 3:1–13.

Zitzler, E. (1999). *Evolutionary Algorithms for Multiobjective Optimization: Methods and Applications*. Ph. D. thesis, Swiss Federal Institute of Technology (ETH) Zurich, Switzerland. TIK-Schriftenreihe Nr. 30, Diss ETH No. 13398, Shaker Verlag, Aachen, Germany.