DESARROLLO DE UN AGENTE INTELIGENTE PARA INTERNET UTILIZANDO ALGORITMOS GENETICOS Y LOGICA DIFUSA*

MARIA JOSE M. BAUTISTA **
FABIO A. GONZALEZ O.***

RESUMEN

Cuando un usuario está buscando información en Internet, recupera una gran cantidad de documentos. En este trabajo, se presenta un Agente Inteligente para recuperación de información que filtra los documentos recuperados y los ordena para mostrar al usuario sólo los que mejor se adaptan a sus preferencias.

Un Algoritmo Genético mantiene el conocimiento sobre las necesidades del usuario y se adapta cuando éstas cambian. Los genes de los cromosomas en la población genética se codifican por medio de funciones de adaptación difusas. Esta codificación permite a cada gen evaluar y ordena los documentos mediante un valor difuso en relación a las preferencias del usuario. La función de adaptación (o de *fitness*) y la frecuencia de aparición de los genotipos difusos dirigen el comportamiento y la capacidad de adaptación del sistema.

El sistema de software que presentamos puede trabajar fuera de línea después de que el usuario obtiene la información de Internet y muestra al usuario solo los documentos más interesantes.

La realización de este artículo se llevo a cabo durante su permanencia en la Escuela de Administración de Negocios durante Septiembre de 1996 a través del programa INTERCAMPUS.

Trabajo realizado en el Departamento de Ciencias de la Computación de la Universidad de Roskilde, Dinamarca. por María José Bautista, Henrik L. Larsen, Jacob Nicolaisen y Torben Svedsen. Será presentado en el FUZZY IEEE '97 a realizarce en Barcelona, España del 1 al 5 de julio.

**Departamento de Ciencias de la Computación e Inteligencia Artificial
Universidad de Granada. 18071, Granada, España
x4269124@turing.ugr.es
***Facultad de Ingeniería, Departamento de Ciencias de la Computación
Escuela de Administración de Negocios. A.A.100888, Santafé de Bogota,Colombia
fgonzal@bacata.usc.unal.edu.co
Profesor e Investigador E.A.N., Colaboró en la realización del Marco Teório del presente artículo y en la adaptación del mismo para ser publicado.

INTRODUCCION

Hoy en día, la expansión de Internet a nivel mundial ha alcanzado medidas inconmesurables. Como consecuencia de esta expansión, también ha crecido la cantidad de información disponible y el número de usuarios con acceso a la misma. La facilidad de conseguir, tan rápido como sea posible, la mejor y más reciente información disponible en la red es la exigencia que impone todo usuario que realiza consultas en Internet tanto a nivel particular, como a nivel empresarial o investigativo. El problema fundamental que se le plantea al usuario que accede a Internet es que la cantidad de información que recupera es tal que desborda su capacidad y tiempo de selección. No toda la información recuperada se adapta a las necesidades del usuario, y el tiempo que éste debe invertir para descartar la información inútil hace que muchas personas recurran a otros medios de búsqueda y recuperación de información.

Un nuevo concepto navegación denominado *Agentes* ha surgido como consecuencia de esta demanda de asistencia en la búsqueda de información. La idea es construir agentes personales de búsqueda que hagan el trabajo sucio, es decir, que lean automáticamente toda la información que el motor de búsqueda de Internet (p. ej. Yahoo, Lycos, AltaVista, etc...) proporciona, desechando aquella información que considere no interesante para un determinado usuario y una determinada consulta, mostrando finalmente al usuario aquellos documentos que ha estimado como los más adecuados según sus necesidades.

En este trabajo, nuestro objetivo va a ser, precisamente, la construcción de uno de estos agentes enmarcándolos en el campo de la Inteligencia Artificial mediante el uso de la Teoría de Conjuntos Difusos y la técnica de optimización denominada Algoritmos Genéticos. Para ser más específicos, la funcionalidad del sistema es la de una herramienta para filtrado de información. Si consideramos el sistema como una caja negra, las entradas serían documentos obtenidos de Internet a partir de una consulta de usuario, y la salida sería solamente una lista ordenada de documentos que el sistema ha seleccionado como mejores a partir de los de entrada y de acuerdo con las preferencias del usuario (Figura 1).

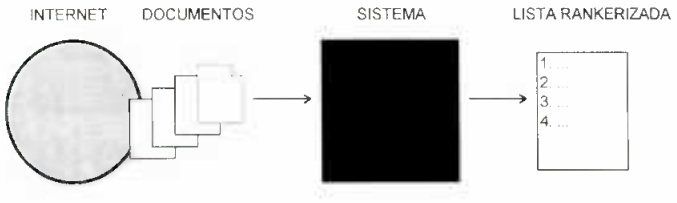


Figura 1: Funcionalidad del Sistema

El sistema debe estar trabajando continuamente en la computadora del usuario, tanto si el usuario está utilizándolo y retroalimentándolo en ese momento como si no. Además, el sistema debe tener la capacidad de suministrar al usuario la nueva información disponible sobre el tema escogido durante un largo periodo de tiempo, en oposición a los sistemas de tipo Consulta-Respuesta, que tratan de maximizar la utilidad para el usuario instantánteamente. La filosofía de la interacción entre el usuario y nuestro sistema subsyace en el hecho de que el usuario es incapaz de expresar las condiciones exactas para las que un documento cubra sus necesidades. Esto es debido al carácter dinámico de las preferencias humanas, de la información disponible y del modelo mental humano que es altamente complejo y, por lo tanto, difícil de comunicar.

Planteamiento del problema e Hipótesis.

El problema planteado en este trabajo es el siguiente: ¿Es posible diseñar un sistema que satisfaga las siguientes necesidades del usuario?

El usuario necesita información en un área específica que le comunica al sistema. El usuario no pregunta directamente al sistema en sí, sino que realiza consultas en Internet. Las necesidades se deben satisfacer por documentos en Internet.

El usuario selecciona los documentos que desea leer proporcionados por el sistema, y suministra *retroalimentación* al sistema evaluando la bondad de adaptación de dichos documentos a sus necesidades. El sistema puede proporcionar nuevos documentos al usuario constantemente.

Las siguientes hipótesis describirán la forma en la que trabajaremos para satisfacer dichos problemas. (Las hipótesis corresponden a los problemas con el mismo número).

El usuario puede comunicar sus necesidades al sistema mediante documentos que expresen sus preferencias.

Una gran cantidad de información está disponible en Internet y está accesible utilizando motores de búsqueda localizados en los servidores.

Se puede construir un módulo de aprendizaje en el que la retroalimentación del usuario se utiliza como indicador del comportamiento del sistema y como guía para el aprendizaje.

El sistema será capaz de estar evaluando documentos constantemente y almacenando la información relevante.

Marco Teórico.

1.- Agentes.

Los agentes inteligentes, como su propio nombre indica, surgieron en el estudio del área de Inteligencia Artificial basada en el comportamiento (*bottom-up*) como alternativa al enfoque clásico de Inteligencia Artificial basada en el conocimiento (*top-down*). Generalmente, todos los agentes son parte actuante de un todo global. En el marco de la computación, un agente se entiende como un programa, por lo que usualmente se utiliza el término de *agente software*. Un agente software debe tener al menos una de las siguientes propiedades:

- *Autonomía*, respecto al usuario, pero orientado hacia el objetivo que éste busca.
- *Asincronicidad*, con respecto al software global en el que está incluido y al sistema operativo.
- *Adaptación*, al medio en el que se encuentra.

Concentrémonos en esta última propiedad: *adaptación*. Una estructura adaptativa (por ejemplo, un trozo adaptativo de software), se caracteriza por una progresiva adaptación a modificaciones repetitivas a través de unos determinados operadores. Por el término progresivo, se entiende una acción orientada hacia un objetivo concreto, que puede ser ganar, sobrevivir o satisfacer las necesidades del usuario. Para analizar la estructura matemática de nuestro futuro agente, de acuerdo al análisis realizado por Holland en 1992, podemos identificar los siguientes elementos:

E : Entorno que rodea la estructura. En nuestro caso, el entorno está formado por redes (Internet), información y el usuario.
τ: Plan adaptativo de la estructura. En nuestro caso, se trata de un Algoritmo Genético que maneja las hipótesis sobre las preferencias de usuario.
μ : El parámetro de éxito de la estructura en cuanto al alcance del objetivo propuesto. En nuestro caso, este parámetro viene dado por la satisfacción del usuario respecto a los documentos conseguidos por el agente hasta ese momento.

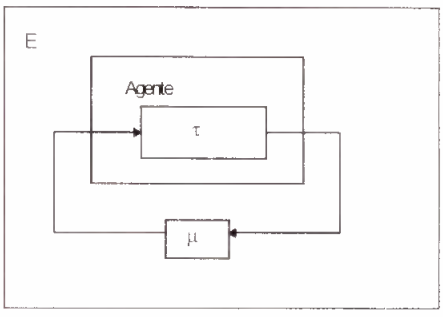


Figura 2 Estructura de un Agente Adaptativo

Tan pronto como el agente software adaptativo se inicializa con una cierta incertidumbre sobre su entorno, su futura adaptación será impredecible y por lo tanto, sus acciones serán autónomas. A causa de esta incertidumbre inicial sobre su entorno, se puede deducir que en algunos casos, el agente software adaptativo no estará preparado para responder síncronamente a su entorno.

En otras palabras, un agente software adaptativo es autónomo y asíncrono y por lo tanto satisface todos los requerimientos necesarios para ser llamado agente software. Nuestro sistema se puede caracterizar como un agente software adaptativo, por lo que a partir de ahora, cuando utilicemos el término agente, nos estaremos refiriendo a agentes software adaptativos.

2.- Algoritmos Genéticos.

El comportamiento adaptativo de nuestro sistema se logra a través del uso de un Algoritmo Genético, el cuál aprende las preferencias del usuario y se adapta al comportamiento cambiante de las mismas.

Los Algoritmos Genéticos [Goldberg 89, Holland 92, Davis 91] se enmarcan dentro de la Computación Evolutiva, un área de las Ciencias de la Computación de gran desarrollo en los años recientes, que junto con la Redes Neuronales constituyen los dos campos representativos de la Computación con Inspiración Biológica, una de las tendencias con más fuerza actualmente y con resultados bastante prometedores.

La Computación Evolutiva se inspira en la evolución biológica, mecanismo que le ha permitido a la naturaleza el desarrollo de seres vivos cada vez más complejos que se adaptan de manera más eficaz a las condiciones del entorno. La Computación Evolutiva simula el proceso evolutivo ya sea para entender mejor los mecanismos naturales de la vida o para solucionar problemas científicos o de ingeniería, como en el caso de los Algoritmos Genéticos.

Los Algoritmos Genéticos aplican el modelo evolutivo a un conjunto de soluciones posibles a un problema dado, a través de dicho proceso se van generando soluciones cada vez mejores, es decir, que resuelven el problema de una manera más satisfactoria.

El Proceso de la Evolución Biológica

Todo ser vivo posee un código genético que determina cada una de los aspectos que lo caracterizan, desde el color de sus ojos y su estatura, hasta su nivel de inteligencia.

Esta información se encuentra codificada en el ADN el cual reside en el núcleo de cada una de sus células, el ADN esta constituido por nucleotidos (cuatro clases diferentes) que se organizan en largas cadenas que codifican secuencias de aminoácidos (veinte clases diferentes) que constituyen las proteínas, las cuales son en últimas los ladrillos con los cuales están construidos los seres vivos. Cada secuencia de nucleotidos que codifican una proteína, que a su vez determina una o más características del ser vivo, es llamada un gen, los genes se encuentran agrupados en cromosomas, el número de cromosomas y genes es una característica particular de cada especie.El mecanismo que permite que una especie perdure en el tiempo es la reproducción, en ella un ser vivo genera un nuevo ser al cual le hereda su código genético, en la mayoría de seres vivos se da un mecanismo de reproducción sexual, es decir, que para engendrar un nuevo ser se necesita de la participación de dos individuos de diferente sexo, el código genético del descendiente es una combinación del código de los padres, a este proceso de combinación se le llama cruce.

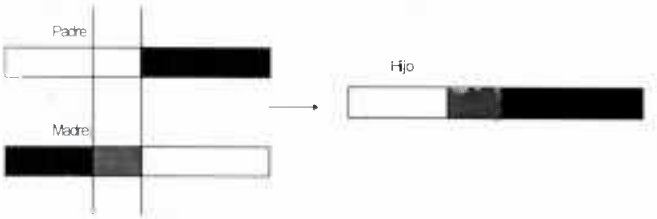


Figura 3 Proceso de Cruce de Código Genético para Generar un Hijo

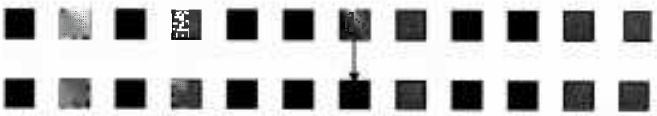


Figura 4 Mutación del Gen 7 de un Cromosoma

La variedad de la información genética de los diferentes individuos de una población, generada por el cruce y por las mutaciones eventuales causadas por agentes externos, implica diferencias en la capacidad de adaptación de cada uno de los individuos, de manera que existen individuos más aptos que estarán en ventaja frente a otros de su misma o diferente especie, en la consecución de alimento, en la conservación de sus vida y en la posibilidad de reproducción; por lo tanto estos individuos tendrán muchas más posibilidades de que su código genético perdure a través de sus descendientes, los cuales heredan su capacidad superior de adaptación. Este mecanismo de supervivencia del más apto se conoce con el nombre de selección natural, y gracias a él, existe un mejoramiento continuo en la capacidad de adaptación de las generaciones sucesivas de una especie.[Darwin 1859]

La Implementación de un Algoritmo Genético

Los Algoritmos Genéticos son una técnica de optimización que permiten maximizar (o minimizar) el valor de una función que depende de un conjunto de variables; el conjunto de los valores posibles que pueden tomar dichas variables es llamado espacio de soluciones. Un Algoritmo Genético explora dicho espacio en búsqueda de soluciones cada vez mejores. Vale la pena aclarar que esta es *una técnica no exacta de optimización*, es decir que no garantiza que la solución que se encuentra es la óptima, sin embargo, las soluciones producidas son bastantes cercanas al valor óptimo, lo cual es suficiente en la mayoría de aplicaciones. La solución de un problema utilizando Algoritmos Genéticos exige desarrollar un mecanismo de codificación que permita representar las diferentes soluciones candidatas. A la codificación de una solución particular se le llama cromosoma, que a su vez está compuesto por genes.

Figura 5 Esquema de una Población de Cromosomas

Un Algoritmo Genético trabaja de la siguiente forma: inicialmente se genera una población de cromosomas, cada uno representa una solución al problema más o menos buena, la función de adaptación se encarga de medir la bondad de cada uno de los cromosomas. Los cromosomas con un valor de adaptación mayor tendrán mayor posibilidad selección.

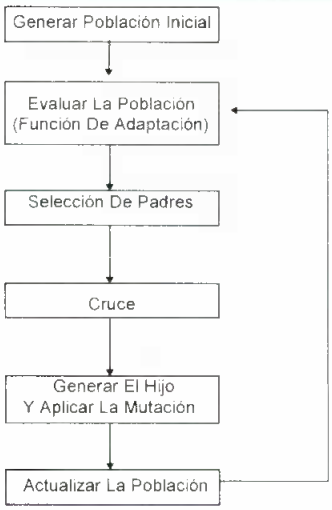


Figura 6 Diagrama de Flujo de un Algoritmo Genético

Una nueva generación de cromosomas se produce a partir de la población actual, aplicando los operadores de selección, cruce y mutación. Se continúa con este proceso de generación de nuevas poblaciones de cromosomas hasta que surja una solución suficientemente buena. En nuestro sistema, un cromosoma codifica, a través de la Teoría de Conjuntos Difusos, una aproximación a las preferencias del usuario, el valor de adaptación de un cromosoma es mayor en la medida que represente de manera más fiel dichas preferencias. La justificación del uso de lógica difusa en los genes de los cromosomas es, precisamente, el reflejo de la vaguedad de la mente humana, es decir, la incapacidad del usuario para expresar sus preferencias de una manera determinística y exacta.

3.- Teoría de Conjuntos Difusos.

Un Mundo Impreciso

En la vida diaria estamos interactuando constantemente con conceptos imprecisos, "está haciendo calor...", "él es una persona alta...", "estamos cerca de allí...", imprecisiones que tienen que ver con la magnitud, la forma, el color, la ubicación, etc. de nuestras percepciones del mundo real. Esta imprecisión la podríamos asociar al desconocimiento o una percepción parcial de la realidad, sin embargo, es algo mucho más profundo, *el mundo es intrínsecamente impreciso*, pues el mundo no lo percibimos en blanco y negro donde las cosas o son o no son, sino como un continuo de grises donde se pasa de ser a no ser de manera imperceptible. Por ejemplo, si tenemos una manzana en nuestra mano estamos seguros de que lo que tenemos en nuestra mano es una manzana, si la mordemos, seguiremos afirmando que es una manzana, si la seguimos mordiendo, pasaremos a un pedazo de manzana y finalmente a nada. ¿En qué momento la manzana paso de serlo a no serlo?. Definitivamente es muy difícil decirlo, cuando tenemos media manzana en nuestra mano, !!tenemos una manzana que a la vez es y no es!!.

Variables Lingüísticas

A través del lenguaje nosotros comunicamos nuestra percepción del mundo, por lo tanto nuestro lenguaje es impreciso; por ejemplo, si nos vamos a referir a la estatura de alguien, generalmente decimos: Juan es alto o Juan es bajo, pero muy pocas veces decimos: Juan mide 1.74 mts. En este caso alto y bajo son valores de la variable Estatura, este tipo de variables se llaman variables lingüísticas y se distinguen de las variables convencionales porque no toman valores exactos, sino valores que expresan un concepto vago como el de ser alto o ser bajo.

¿Cómo se relacionan las variables lingüísticas con las variables convencionales?, es decir, ¿cómo se relaciona ser Alto con tener una estatura específica?. La gran mayoría de personas estarían de acuerdo en afirmar que alguien que midiera 1.90 mts., es una persona alta pero, ¿qué pasaría con alguien que midiera 1.70 mts?. Algunos dirían que es Alto, otros dirían que tiene una estatura promedio y otros tal vez dirían que es bajo. Sin embargo, la gran mayoría (en nuestro medio) estaría de acuerdo con una clasificación como la siguiente:

Altos = {los X, tal que la estatura de X es mayor o igual a 1.70 mts }
Promedio = {los X, tal que la estatura de X es mayor o igual 1.60 y menor a 1.70 mts }
Bajos = {los X, tal que la estatura de X es menor a 1.60 mts }

Esta clasificación, aunque refleja en cierto modo la idea intuitiva que tenemos de Alto, Bajo y Promedio, tiene un grave inconveniente. ¿Qué pasaría con una persona que midiera 1.699 mts y alguien que midiera 1.70 mts? la primera se clasificaría como Promedio y la segunda como Alto, lo cual no tiene mucho sentido pues solo difieren en un milímetro.

Se podrían proponer como soluciones bajar o subir los límites, o aumentar el número de categorías, pero esto no solucionaría el problema. Definitivamente la Teoría de Conjuntos convencional no permite modelar la concepción que manejamos de Alto, Promedio o Bajo. El problema reside en el hecho que la Teoría de Conjuntos exige que un elemento cualquiera, debe pertenecer o no pertenecer a un conjunto dado, lo cual implica que una persona es alta o no es alta, pero para nosotros no es así, hay personas Altas y otras menos Altas, es decir, hay grados de pertenencia intermedios entre el pertenecer y no pertenecer al conjunto de los Altos.

Los Conjuntos Difusos

La siguiente gráfica muestra una mejor aproximación al concepto de Alto, a través de una función que asocia a cada estatura un grado de pertenencia al conjunto de las personas Altas:

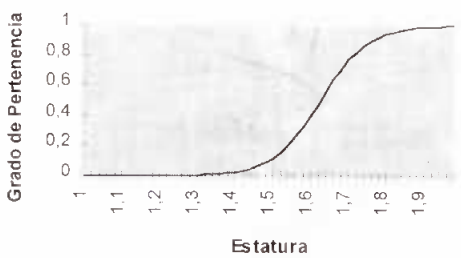


Figura 7: Función de Pertenencia al conjunto Altos.

En este caso una persona con una estatura de 1.50 mts. tendría un grado de pertenencia de 0.1 al conjunto de los altos, una persona con 1.90 mts. tendría un grado de pertenencia de 0.98 y una persona de 1.64 mts. tendría un grado de pertenencia de 0.55. Esta noción coincide de una manera más fiel con la noción intuitiva que tenemos de una persona Alta.

El anterior ejemplo es una muestra de un Conjunto Difuso, es decir un conjunto donde la función de pertenencia es una función continua entre 0 y 1, no como en el caso de los conjuntos convencionales, donde la función de pertenencia es booleana, es decir, que solo toma dos valores el 0 o el 1, pero nunca un valor intermedio.

La noción de conjunto difuso permite relacionar de una manera más precisa los valores de una variable lingüística con los respectivos valores de una variable convencional, lo cual nos facilita representar de una manera más formal la percepción humana del mundo, que es evidentemente imprecisa. Por ejemplo, con esta noción podemos dar un sentido mucho más preciso a sentencias como: "si el motor está caliente disminuya un poco la entrada de combustible", esto hace más fácil la comunicación con sistemas computacionales, los cuales necesitan ordenes exactas, en un lenguaje mucho más cercano al que usamos en la vida diaria.

La Lógica Asociada a los Conjuntos Difusos

La Teoría de Conjuntos convencional esta asociada a una lógica bivalente de primer orden, determinada por las funciones de pertenencia que solo pueden tomar dos valores: pertenece o no pertenece, 1 o 0, falso o verdadero. Esta lógica, es la lógica clásica y cada una de sus operaciones corresponde a operaciones entre conjuntos, por ejemplo, a partir del concector lógico Y ($\wedge$), el cuál indica que **a**$\wedge$**b** es verdadero sí y solo sí tanto **a** como **b** son verdaderos, podemos definir la intersección de la siguiente manera :

Sean A y B conjuntos, $\mu_A[x]$ y $\mu_B[x]$ sus funciones de pertenencia definidas así:

$$\mu_A[x]=\begin{cases}1 \text{ si } x\in A\\ 0 \text{ si } x\notin A\end{cases} \qquad \mu_B[x]=\begin{cases}1 \text{ si } x\in B\\ 0 \text{ si } x\notin B\end{cases}$$

entonces

$$\mu_{A\cap B}[x]=\mu_A[x]\wedge\mu_B[x],$$

a partir de la función de pertenencia podemos obtener el conjunto A $\cap$ B así,

$$A\cap B=\{x, \text{ tal que } \mu_{A\cap B}[x]=1\}=\{x, \text{ t.q. } \mu_A[x]=1 \text{ y } \mu_B[x]=1\}=\{x, \text{ t.q. } x\in A \text{ y } x\in B\}$$

Análogamente se puede definir las funciones de pertenencia y conjuntos para los operadores v,~,→, ... de la lógica de primer orden. De la misma forma en que los Conjuntos Convencionales están asociados a la lógica clásica bivaluada, los Conjuntos Difusos están asociados a una lógica multivaluada, es decir a una lógica donde los valores de verdad son los infinitos números reales entre 0 y 1, esta lógica fue desarrollada por Zadeh [Zadeh 65], y recibe el nombre de *Lógica Difusa*.

Operadores Difusos

Al igual que en los conjuntos convencionales, en los conjuntos difusos se pueden definir operaciones como complemento, unión e intersección. Zadeh definió estos operadores de la siguiente manera:

Supongamos que tenemos dos conjuntos difusos A y B, cuyas correspondientes funciones de pertenencia son $\mu_A[x]$ y $\mu_B[x]$ respectivamente, entonces:

$\mu_{A\cup B}[x]$ = Máximo($\mu_A[x]$, $\mu_B[x]$)	Unión
$\mu_{A\cap B}[x]$ = Mínimo($\mu_A[x]$, $\mu_B[x]$)	Intersección
$\mu_{\sim A}[x]$ = 1 - $\mu_A[x]$	Complemento

Considerando de nuevo el conjunto difuso Altos que definimos anteriormente, podemos definir con su ayuda el conjunto de los Bajos de la siguiente forma:
Bajos = ~Altos, graficando $\mu_{\sim Altos}[x]$:

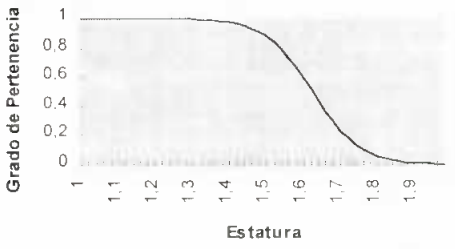


Figura 8: Función de Pertenencia al Conjunto Bajos.

De la misma manera podemos definir el conjunto difuso de personas con estatura promedio por: Promedio = Bajos $\cap$ Altos, representando gráficamente $\mu_{\sim Altos\ Altos}[x]$ obtenemos: (en este caso se normalizó el resultado de manera que el mayor valor correspondiera a 1)

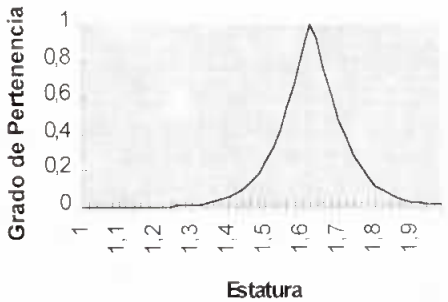


Figura 9: Función de Pertenencia al Conjunto Promedio.

4.- Razonamiento en Inteligencia Artificial.

En el campo de la Inteligencia Artificial (IA) se ha estado [cita...] estudiando sobre la denominada *Hipótesis de razonamiento en mundo cerrado*. Esta hipótesis asume que en el sistema de IA que se estudia, la base de su conocimiento es una representación completa del dominio del problema. En la práctica, esta hipótesis implica que el contacto de los sistemas de IA con el dominio del problema es indirecto y solo se realiza a través de un experto humano (en el área específica del problema). El experto debe traducir la información desde el dominio del problema de modo que el sistema (construido en lógica de primer orden) no tenga que tratar con ningún tipo de concepto difuso, vago o contradictorio. Esta aproximación tradicional de IA se denomina *basada en el conocimiento* puesto que la inteligencia consiste en la capacidad del sistema para obtener conclusiones a partir de una base de conocimiento expresada en lógica de primer orden. Por lo tanto, los sistemas basados en el conocimiento son normalmente construidos en un lenguaje descriptivo consistente en reglas, hechos y excepciones, que los hacen ideales para el manejo de bases de datos.

Sin embargo, nuestro sistema no queda caracterizado en esta aproximación, sino en los sistemas *basados en el comportamiento*. La justificación a esta elección se encuentra en la diferencia de capacidad de razonamiento en unos y otros sistemas. La principal diferencia de estos sistemas con los basados en el conocimiento está en que el enfoque de atención no se encuentra en el conocimiento exacto, pero sí en una mayor y más directa información con el dominio del problema.

La modelización de agentes es el núcleo central de este último tipo de razonamiento. Un sistema de IA basado en el comportamiento se construye en un lenguaje procedural para el almacenamiento y acceso de datos a través de probabilidades. Esta aproximación probabilística tiene la ventaja de que se puede introducir un grado de adaptación en el sistema. De este modo, no solo los datos pueden tener origen difuso, sino que el acceso y el efecto de los datos individuales son una consecuencia de la estructura global adaptativa del sistema en un punto determinado del tiempo.

El proceso de razonamiento se puede dividir en *abducción*, *deducción* e *inducción*. La deducción, utiliza reglas generales y hechos conocido para predecir o confirmar algo. La abducción calcula las causas a partir del conocimiento sobre reglas generales y los resultados observados. La inducción es procesar las reglas generales a través de múltiples observaciones (por ejemplo, causas y resultados).

Razonamiento basado en el Comportamiento

El razonamiento basado en el comportamiento en Inteligencia Artificial tiene sus orígenes en las investigaciones de adaptación y evolución. Uno de los principales investigadores en este campo fue C.S. Pierce (1863-18..), el cuál construyó una teoría que situaba el descubrimiento entre lo aleatorio y lo determinístico y realizó una analogía entre la teoría darwiniana y el comportamiento:

«...en la evolución de la ciencia, el descubrimiento juega el mismo papel que las variaciones en la reproducción toman en la evolución de formas biológicas, de acuerdo a la teoría darwiniana»[1]

Si en nuestro sistema basado en el comportamiento, tuviésemos que aprender de un modo aleatorio, tendríamos que calcular (decisiones, acciones y efectos) durante millones de años a causa del enorme espacio de posibilidad. Sin embargo, si aprendiésemos deterministiamente, no aprenderíamos del todo. Lo mismo ocurre en el comportamiento humano[2].

El descubrimiento que nuestro sistema realiza sobre los usuarios se produce a través de los *operadores genéticos*, los cuales parten de hipótesis previas para construir nuevas que las substituyan. Los operadores genéticos aseguran que el descubrimiento es un proceso entre la aleatoriedad y la determinación.

La elección de un sistema basado en la evolución refleja la afirmación de que el usuario razona tal y como lo asume Pierce. Esto significa que el usuario debería poder pensar y leer de manera inquisitiva y que **no es capaz de conocer exactamente sus preferencias a causa del carácter evolucionario de su conocimiento**. Se podría decir que el usuario solo puede expresar sus preferencias en el pasado, y esta es la razón por la cuál nuestro sistema sólo puede contener hipótesis.

Para manipular estas hipótesis mediante operadores genéticos, las hipótesis deben ser equivalentes a cromosomas conteniendo genes de diferentes tipos. Cada gen contiene un término y un valor numérico. El término (proveniente de los documentos tomados de la red que el sistema procesa) debe estar presente en un nuevo documento que es evaluado por dicho gen. El resultado de esta evaluación es un valor difuso. Este valor depende de una función difusa específica del tipo al que pertenezca el gen. La evaluación de un documento por parte de los cromosomas (hipótesis), es la media de las evaluaciones de los genes.

[1] Cita de Pierce tomada de Anderson(1987), pág.47

[2] Anderson (1987), pág.47

Cabe resaltar que nuestro sistema no contiene hechos ya que los téminos sólo son parte de hipótesis que están cambiando continuamente. La validación de una hipótesis individual es una función de la validación del resto de las hipótesis. Esto significa que tenemos una regresión sin fin respecto a la validación de una hipótesis. Esto, obviamente, no es un hecho en un sentido descriptivo, pero teóricamente debería hacer razonar nuestro sistema basado en el comportamiento del siguiente modo:

- *Abducción* : A través de la retroalimentación del usuario, el sistema conoce la bondad de los documentos recuperados. Una mala retroalimentación produce nuevas hipótesis creadas a partir de información de antiguas y el sistema debe ser capaz de filtrarlas y detectar qué hipótesis funcionan mejor para darles una mayor probabilidad de conseguir alguna de la información representada en las nuevas hipótesis.

- *Deducción* : Este proceso es el que se adapta más a la capacidad computacional de las máquinas. Se podría decir que la deducción es hacer cálculos, mientras que la abducción es crear funciones para hacer cálculos. En nuestro caso, la deducción es calcular la puntuación de los nuevos documentos.

- *Inducción* : A través de sucesivas evaluaciones de la adaptación (fitness) de las hipótesis individuales, la validación de las hipótesis se comprueban inductivamente: cuanto mayor sea el fitness, mayor plausibilidad tendrá la hipótesis. El fitness se utiliza para la abducción, pero también controla el impacto global de todas las hipótesis en la colección de todos los documentos.

Por lo tanto, se pretende que la representación de hipótesis mediante cromosomas permitirá a nuestro sistema abducir, deducir e inducir a la manera de Peirce.[Op.Cit]

Descripción del Sistema.

El sistema trabaja sobre la información (por ejemplo, documentos HTML) que un motor de búsqueda (Alta Vista, en nuestro caso) recupera de Internet y está compuesto por diferentes módulos: el *módulo de términos*, el *módulo de aprendizaje* y el *módulo de evaluación*.

El *módulo de términos* transforma cada documento en un árbol binario que almacena estadísticas sobre los términos relevantes del documento.

El *módulo de aprendizaje* es el objeto central del sistema. Este módulo debe ser capaz de ajustar la representación de los datos a las preferencias del usuario de modo que dichos datos sean consistentes con los últimos valores de retroalimentación que el usuario haya suministrado al sistema, y todavía retenga los datos esenciales históricos sin incrementar el tamaño total de los datos sobre las preferencias del usuario. Este conocimiento se mantiene en la población de cromosomas de un Algoritmo Genético (AG). Cada *cromosoma* es una hipótesis sobre como evaluar un documento y todos los cromosomas compiten para predecir la adaptación de un documento a las preferencias del usuario. La evaluación que los cromosomas hacen de un documento se denomina *puntuación del cromosoma* y la capacidad de un cromosoma de clasificar un documento se denomina *fitness* o grado de adaptación del cromosoma. Un cromosoma contiene *genes* que se representan como términos y funciones de pertenencia difusas obtenidos a partir de los documentos. Por lo tanto, cada término puede estar contenido en uno o más genes con un valor ideal (valor modal del conjunto difuso) relacionado con las ocurrencias del término en el documento.

Finalmente, el *módulo de evaluación* asigna una *puntuación* a los documentos utilizando la información presente en el módulo de aprendizaje. Un esquema general de la arquitectura del sistema es:

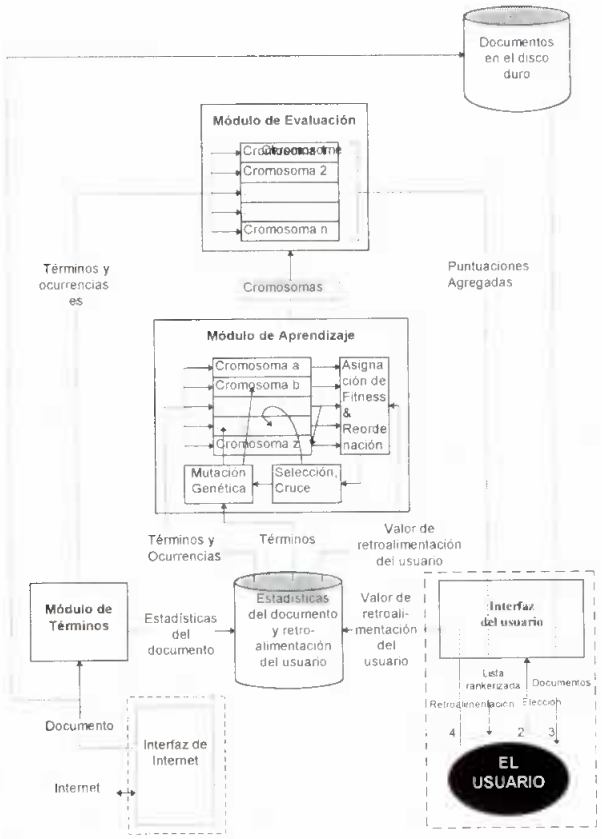


Figura 10: Arquitectura del Sistema.

Cuando el usuario lee un documento retroalimenta al sistema mediante un valor que exprese su satisfacción respecto a sus preferencias. Dicho valor podría ser expresado, por ejemplo, escogiendo entre diferentes botones en la *interfaz de usuario* que indiquen la utilidad del documento como "pobre", "moderado", "bueno" y "muy bueno" [Yager 96], que no son sino etiquetas linguísticas que serán defuzzificadas en un valor numérico procesable. Mediante esta misma interfaz, se presenta al usuario una lista de documentos rankerizados con información tal como, por ejemplo, el título y las dos primeras líneas de cada documento. Finalmente, una *interfaz de Internet* formula consultas y los manda a un motor de búsqueda externa (cualquiera de los disponibles en Internet). La funcionalidad de la interfaz de Internet es proporcionar al módulo de evaluación con los documentos que satisfagan los requerimientos de la consulta que el usuario efectuó al sistema.

Combinando Algoritmos Genéticos y Teoría de Conjuntos Difusos en el Módulo de Aprendizaje.

Los AG son técnicas de optimización que dado un conjunto de parámetro $\mathbf{p}_i$ tratan con una función de aproximación. En nuestro caso, esta función representa las preferencias de un usuario buscando información en Internet.

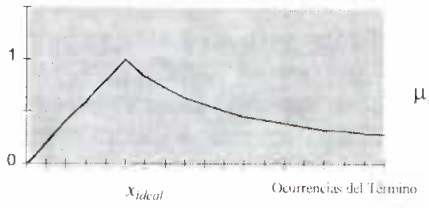
$$\mathbf{F(p_1, \ldots, p_n) = Preferencias\ del\ Usuario}$$

El usuario expresa sus preferencias mediante un documento o conjunto de documentos inicial a partir del cual el sistema consigue información sobre tópicos, ocurrencias de términos, etc. que permite al sistema encontrar nuevos documentos más cercanos a las necesidades del usuario. El usuario (y a veces el sistema), puede evaluar nuevos documentos y retroalimentarlo, de modo que el sistema puede aprender sobre las preferencias del usuario.

Codificación difusa de los Genes

Los AG requieren que el conjunto de parámetros (solución) estén codificados en un cromosoma compuesto por *genes*. Cada gen pertenece a un alfabeto específico, generalmente el alfabeto binario. Sin embargo, en nuestro sistema, la codificación es especial en cuanto a que debe representar las preferencias del usuario. Cada gen indica una preferencia del usuario mediante la atribución de un término a una función de pertenencia difusa [Zadeh 65, Zadeh 77]. Hay cuatro funciones estándares denominadas genotipos. Mediante G_μ representaremos la *evaluación de un gen de un* documento, es decir, la evaluación de la función de pertenencia del gen, sea cual sea su genotipo.

Gen de tipo 1: La información de esta clase de gen representa el número de ocurrencias f de un término que al usuario le gusta. El extremo de la función x_{ideal}, es el número 'ideal' de veces que el término aparece en el documento. La función de pertenencia de este tipo de gen tiene la siguiente forma:

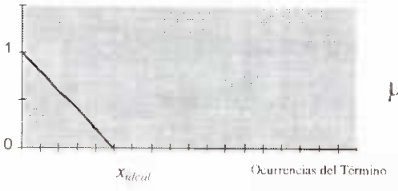


$$\mu_{=}(x) = \begin{cases} \dfrac{x}{x_{ideal}} & x < x_{ideal} \\ \dfrac{x_{ideal}}{x} & x \geq x_{ideal} \end{cases}$$

Figura 11: Función de Pertenencia Gen de Tipo 1

El efecto de la forma de la función es que el término representado tiene debe aparecer cercano al ideal para influenciar sobre la evaluación del cromosoma con el máximo grado.

Gen de tipo 2: Este tipo de gen contiene información sobre el número de ocurrencias de un término que el usuario estima no debía ser alcanzado por dicho término en el documento 'ideal'. La función de pertenencia de este tipo de gen tenía la siguiente forma:

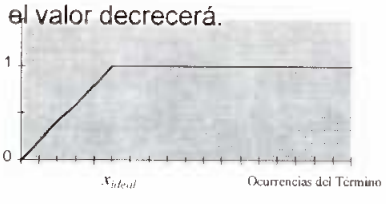


$$\mu_{\leq}(x) = \begin{cases} \dfrac{x_{ideal} - x}{x_{ideal}} & x < x_{ideal} \\ 0 & x \geq x_{ideal} \end{cases}$$

Figura 12 Función de Pertenencia Gen de Tipo 2

La funcionalidad de este tipo de gen es para los términos que el usuario no quiere que aparezca.

Gen de tipo 3: Este gen representa el número menor de ocurrencias que el término debe tener en el documento que obtiene el máximo valor. Para menos apariciones, el valor decrecerá.

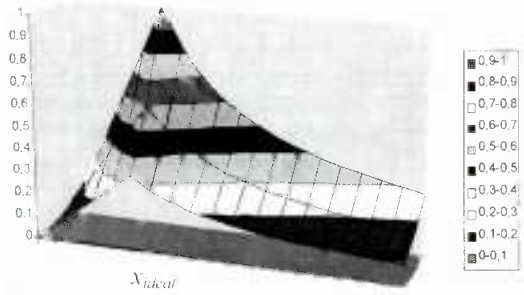


$$\mu_{\geq}(x) = \begin{cases} \dfrac{x}{x_{ideal}} & x < x_{ideal} \\ 1 & x \geq x_{ideal} \end{cases}$$

Figura 13 Función de Pertenencia Gen de Tipo 3

La funcionalidad de este gen es encontrar los términos que el usuario considera muy adecuados, por lo que la forma de la función es constante desde el ideal hasta el infinito.

Gen de tipo 4: Este tipo de gen representa la parte del documento donde un término debe aparecer. Si dividimos el documento en tres partes, el término puede tener asociado uno de los tres valores dependiendo de la parte del documento en la que aparezca. De este modo, si el término aparece en el primer 10% del documento, el peso del término será w_1, si aparece en el 80% central del documento, entonces el peso será w_2 y si aparece en último 10%, entonces tomará el valor w_3 . En los experimentos realizados, se han tomado los valores (w_1 , w_2 , w_3) = (0.99, 0.33, 0.66). Los valores resultantes de los términos con su peso se agregan eligiendo el máximo de ellos. En realidad, esto es una agregación de los genes de tipo 1, así que la función de pertenencia es:



$$\mu_{\equiv}(\vec{x}) = \max_{j}(\mu_{=}(x_j) \cdot w_j)$$

$i = 1, \ldots,$ *número de partes en el documento*

Figura 14 Función de Pertenencia Gen de Tipo 4

Un gen es una tripleta (t_i , x_i , $(x_{ideal})_i$), donde t_i identifica el tipo de gen, x_i es un término, y $(x_{ideal})_i$ es el número ideal de ocurrencias asociadas al gen. En un mismo cromosoma, pueden aparecer dos genes del mismo tipo con el mismo número ideal de ocurrencias. La representación de un cromosoma podría ser:

Gen de tipo t_1	Gen de tipo t_2	Gen de tipo t_3	...	Gen de tipo t_n
x_1	x_2	x_3	...	x_n
$(x_{ideal})_1$	$(x_{ideal})_2$	$(x_{ideal})_3$	...	$(x_{ideal})_n$

Figura 15 Estructura de un cromosoma.

Operadores Genéticos

Los operadores genéticos son básicamente tres: la *selección* se ocupa de la elección de los cromosomas de la población que se reproducirán. El *cruce* toma secuencias de gens de cada cromosoma seleccionado y los combina para obtener descendencia. El cruce es necesario para mantener la diversidad en la población (explotación). La *mutación* es la alteración aleatoria de un gen en el cromosoma seleccionado. La mutación es necesaria para prevenir la convergencia prematura hacia los óptimos locales para garantizar que se recorre todo el espacio de búsqueda (exploración) [Goldberg 89] .

Selección

Existen diferentes estrategias de selección, pero en la mayoría de ellas, los cromosomas con mayor fitness tiene una mayor probabilidad de producir descendencia para las futuras generaciones que aquellos cromosomas con un fitness menor. En nuestro sistema, tienen lugar dos selecciones. Por un lado, el cromosoma para llevar a cabo el cruce (padre) se selecciona aleatoriamente cada vez entre los primeros *x* cromosomas de una población ordenada decrecientemente según el fitness. Para la madre, se selecciona otro cromosoma aleatoriamente entre la población comprendida entre la primera posición y la posición del padre. Esto hace que los cromosomas con un mayor valor de fitness tengan una mayor probabilidad de convertirse en padres. Por otro lado, solo un cromosoma se selecciona cada vez para mutar. Dicho cromosoma es seleccionado aleatoriamente entre la población con una probabilidad de (1/longitud del cromosoma).

Cruce

Tras la selección, las cadenas son copiadas y el cruce ocurre sobre las copias. Cada par de estas copias realiza el cruce del siguiente modo: cada posición de un gen en el cromosoma tiene una probabilidad de 0.2 de ser un punto de cruce. Cuando todos los puntos de cruce han sido determinados, los genes situados entre cada dos de dichos puntos se intercambian entre los dos padres. El nuevo cromosoma generado, el hijo, remplazará el peor cromosoma de la población actual. Este tipo de cruce descrito es un caso general del operador de cruce multi-punto, donde el número de puntos de cruce para cada dos cromosomas es, como máximo, la longitud del cromosoma.

Mutación

En el código binario, la mutación cambia el 0 por el 1 o el 1 por el 0. En nuestro sistema, la mutación sobre el gen tiene lugar del siguiente modo: el tipo del nuevo gen es el mismo que el del gen anterior, el término cambia introduciendo un nuevo término procedente del documento actual, y el número ideal para dicho término será el número de ocurrencias del término en el documento de donde se ha obtenido. Por si misma, la mutación es un proceso de recorrido aleatorio a través del espacio de búsqueda del problema. La frecuencia de mutación, según estudios empíricos y en analogía con la biología, debe ser del orden de 0.0001 [Holland 92 ; sin embargo, en nuestro sistema, este valor se debe incrementar para introducir una mayor diversidad en la población.

De todos modos, esta probabilidad depende de la longitud del cromosoma, de forma que cuanto más genes tenga el cromosoma, menor será la probabilidad de mutación de un gen.

Función Fitness

La función fitness calcula la adaptación de cada cromosoma de la población. Normalmente, el resultado del proceso de evolución a través de una generación es una nueva generación en la que los cromosomas con mayor fitness aparecen con más frecuencia que los de menor fitness. Como resultado de esto, algunas veces la población pierde diversidad ya que muchas copias del mismo cromosoma aparecen los cromosomas con menor fitness mueren rápidamente.

Esto representa una desventaja en nuestro sistema debido a que la entrada a la función de adaptación son documentos y puntuaciones actualizadas cada vez que el usuario interactua con el sistema. Es decir, como estos parámetros no están fijos, sino que son dinámicos, los cromosomas que en un principio pueden ser malos, se podrían convertir en buenos en futuras generaciones. Por lo tanto, debemos permitir a los cromosomas que potencialmente tienen una buena combinación de genes sobrevivir a través de varias generaciones.

La forma más eficiente de conseguir esto es mantener el fitness de cada cromosoma (C_f) a través de varias generaciones, pero modificándolo mediante la suma de un *payoff* (C_p) (incremento) para incrementar así el fitness acumulado de cada cromosoma, y restarle una cierta *lifetax* (C_l) (penalización). El payoff se debe calcular mediante la relación entre la evaluación del documento por parte del cromosoma y la retroalimentación que el usuario da al sistema sobre dicho documento, de forma que, cuanto menor sea la distancia entre dichos valores, mayor será el payoff que obtiene el cromosoma. La lifetax podría ser el fitness mínimo de la población.

$$C_{f_i} = \begin{cases} C_{f_{i-1}} + C_{p_i} - C_{l_i} & i = 1, 2, \ldots, \text{número de generaciones} \\ C_{p_i} - C_{l_i} & i = 0 \end{cases}$$

donde,

$$C_{p_i} = 1 - \left| C_\mu - U \right| \qquad (1)$$

donde U es la evaluación del documento por parte del usuario y la evaluación del documento por parte del cromosoma (C_μ) se calcula como la media de la evaluación del documento por parte de los genes contenidos en dicho cromosoma:

$$C_\mu = \frac{1}{n} \cdot \sum_{i=1}^{n} (G_\mu)_i$$

Mediante esta forma de asociar los fitness, conseguimos una mayor diversidad en la población, y una mejor supervivencia de los cromosomas más débiles a través de las generaciones.

Para mantener una mayor diversidad en la población, también prohibimos que aparezcan dos cromosomas iguales en la población (normalmente, un cromosoma puede aparecer varias veces en la población). Cuando los nuevos cromosomas se incluyen en la población provenientes de la mutación y el cruce, son inicializados con un fitness que es la mitad del máximo fitness presente en ese momento en la población. El payoff y la lifetax para dichos cromosomas no se calcula hasta la siguiente generación, de modo que aparecen en el centro de la población, reemplazando los cromosomas con menor fitness.

Test del Sistema.

Para probar la bondad del sistema, se deben analizar dos características:

- **Precisión en la Predicción**: Es decir, la habilidad de, tan exacto como sea posible, predecir la retroalimentación del usuario. Para probar el sistema, debemos exponerlo a un número determinado de situaciones que revelen su capacidad para razonar y aprender de forma que sea capaz de descubrir las similaridades entre los documentos a pesar de sus aparentes diferencias y de clasificar nuevos tipos de documentos que reflejen la evaluación del usuario.

- **Capacidad de Recapitulación**: La capacidad de, con precisión y rapidez, ajustarse a los cambios en el entorno (cambio de las preferencias del usuario y de la información disponible).

Conclusión.

El problema planteado en las consultas de Internet está latente en cada uno de los usuarios de la red. El sistema presentado en este trabajo constituye una solución sencilla y eficaz, que no depende del tipo de usuario ni de el motor de búsqueda que se utilice en la consulta de Internet.

Por otro lado, el sistema puede trabajar tanto dentro como fuera de línea, puede aprender el perfil de un determinado usuario y puede adaptarse al cambio de preferencias del mismo.

En futuros trabajos, se podría introducir un controlador difuso para los parámetros del algoritmo genético tales como la probabilidad de cruce, de mutación y el tamaño de la población. También se podrían introducir nuevos tipos de genes reconocedores de patrones de gráfico, video o voz.

Referencias.

Anderson, D.R. (1987*), "Creativity and the Philosophy of C.S. Peirce"*

Davis, L. (1991), *"Handbook of Genetic Algorithms"*, Van Nostrand Reinhold.

Darwin, Charles. (1859), *"On the origen of species by means of natural selection or the preservation of favored races in the struggle for life"*, Murray.

Goldberg D.E. (1989), *"Genetic Algorithms in Search, Optimization and Machine Learning"*, Addison-Wesley.

Holland, J.H., Holyoak, Nisbett & Thagard (1986), Ò*Induction*Ó, MIT Press.

Holland, J.H. (1992), "*Adaption in Natural and Artificial Systems*", MIT Press.

Larsen, Henrik L. and R.R. Yager (1996), "*Query Fuzzification for Internet Information Retrieval*", Datalogiske skrifter, No. 60, R.U.C. Roskilde.

Maes, Pattie (1995), "*Modeling Adaptive Autonomous Agents*" in C.G. Langton (1995) "Artificial Life", MIT Press.

Yager, Ronald (1988), "*On Ordered Weighted Averaging Aggregation Operators in Multicriteria Desicionmaking*", IEEE Trans. Systems, Man and Cybernetics, Vol 18, No 1.

Yager, Ronald (1996), "*Intelligent Agents on the World Wide Web*" in "Flexible Query Answering Systems" ed. Christiansen, H., H.L. Larsen & T. Andreasen, Datalogiske skrifter No. 62, R.U.C. Roskilde.

Zadeh L.A. (1965), "*Fuzzy Sets*", Information and Control, 83, 338-353.

Zadeh L.A. (1977), "*Theory Fuzzy Sets*" in J. Belzer, A. Holzman, A. Kent, "Encyclopedia of Computer Science and Technology". Marcel Dekker.