TIAGO FREIRE GARCIA

PROPOSTA DE UMA MÁQUINA DE BUSCA EFICIENTE PARA DOCUMENTOS NA WEB USANDO LÓGICA FUZZY

Monografia de Graduação apresentada ao Departamento de Ciência da Computação da Universidade Federal de Lavras como parte das exigências da disciplina de Projeto Orientado II para a obtenção do título de Bacharel em Ciência da Computação.

Orientadora

Profa. Olinda Nogueira Paes Cardoso

LAVRAS
MINAS GERAIS - BRASIL
2002

TIAGO FREIRE GARCIA

PROPOSTA DE UMA MÁQUINA DE BUSCA EFICIENTE PARA DOCUMENTOS NA WEB USANDO LÓGICA FUZZY

Monografia de Graduação apresentada ao Departamento de Ciência da Computação da Universidade Federal de Lavras como parte das exigências da disciplina de Projeto Orientado II para a obtenção do título de Bacharel em Ciência da Computação.

APROVADA em __ de _____ de _____.

_____
Prof. André Luiz Zambalde

_____
Profa. Olinda Nogueira Paes Cardoso
(Orientadora)

_____
Prof. Joaquim Quintero Uchôa

LAVRAS
MINAS GERAIS – BRASIL
2002

Dedico este trabalho a todos que me ajudaram,
à minha família, em especial aos meus pais,
porque se não fosse por eles eu não estaria aqui.

RESUMO

Palavras-chave: recuperação de informação, Web, lógica fuzzy, máquina de busca.

Este trabalho apresenta um estudo sobre Recuperação de Informação, dando ênfase maior em máquinas de busca, lógica Fuzzy e o uso da lógica Fuzzy em Recuperação de Informação. Em seguida, é proposta um máquina de busca para documentos na Web usando lógica Fuzzy. No final são apresentadas algumas conclusões e algumas possibilidades para trabalhos futuros.

ÍNDICE

LISTA DE FIGURAS V

LISTA DE TABELAS VI

CAPÍTULO 1 - INTRODUÇÃO 1

CAPÍTULO 2 – RECUPERAÇÃO DE INFORMAÇÃO 3

2.1. A tarefa do usuário 4
2.2. Visão lógica dos documentos 5
2.3. Passado, Presente e Futuro 5
2.4. O Processo de Recuperação 6
2.5. Máquinas de Busca 8
2.5.1. Arquitetura Centralizada 8
2.5.2-Arquitetura Distribuída 8
2.5.3. Interfaces com o usuário 9
2.5.4. Ordenação (*Ranking*) 10
2.5.5-Índices 10
2.5.6 Modelo Booleano 11

CAPÍTULO 3 – LÓGICA FUZZY 13

3.1 Introdução à lógica 13
3.2 Introdução à Lógica Fuzzy 13
3.3 Conjuntos Fuzzy 15

CAPÍTULO 4 – INTEGRAÇÃO ENTRE LÓGICA FUZZY E RECUPERAÇÃO DE INFORMAÇÃO 20

4.1 Uma nova aproximação a sistemas de recuperação de informação usando expressões Fuzzy 22
4.1.1 Noções Básicas 22
4.1.2 Descrição de um modelo corrente de recuperação de documento 26
4.2 Modelo Fuzzy versus Modelo Probabilístico para julgamento de relevância do usuário 29
4.2.1 Aproximação 30
4.2.2 O modelo probabilístico 30
4.2.3 Modelo Fuzzy 32
4.2.4 Desenvolvimento de hipótese e experimento 33
4.2.5 Análise 35
4.2.6 Resultados 37

4.3 Um sistema de recuperação de informação usando matriz de conexão de palavras-chave e um método de aprendizado 38
4.3.1 Matriz de conexão de palavras-chave 39
4.3.2 Recuperação Fuzzy usando a matriz de conexão de palavras-chave 39
4.3.2.1. Revisão do método de recuperação Fuzzy 40
4.3.2.2 Computação do valor de afinidade usando matriz de conexão de palavras-chave 41
4.3.2.3 Geração de índices Fuzzy 42
4.3.2.4 Computação do valor de relevância de cada sub-consulta 42
4.3.2.5 Computação do valor de relevância total 43
4.3.3 Refinamento da consulta 43
4.3.4 Método de aprendizado da matriz de conexão de palavras-chave 44
4.3.5 Avaliação do desempenho 44
4.3.5.1. Modelo de avaliação 44
CAPÍTULO 5 – PROPOSTA DE UMA MÁQUINA DE BUSCA EFICIENTE PARA DOCUMENTOS NA WEB USANDO LÓGICA FUZZY 47
CAPÍTULO 6 – CONCLUSÕES 51
6.1 Trabalhos Futuros 51
REFERÊNCIAS BIBLIOGRÁFICAS: 53

LISTA DE FIGURAS

Figura 1: O processo de recuperação de informação 7

Figura 2: Pertinência em conjuntos Fuzzy 15

Figura 3: Exemplo com o grau de altura representando pertinência 16

Figura 4: Lista de consultas usadas.....35

Figura 5: Proposta de uma máquina de busca para Web. 49

LISTA DE TABELAS

Tabela 1: Relação que descreve os documentos $d \in D$. 27

Tabela 2: Relação 0.2-level $F_{(0.2)}$ 27

Tabela 3. Formula usada com consultas complexas 37

Tabela 4. Comparação do método proposto usando matriz de conexão de palavras-chave e o método crisp. 45

CAPÍTULO 1
INTRODUÇÃO

Atualmente, a *World Wide Web* (WWW ou Web), tem se caracterizado como um dos maiores mecanismos de disseminação de informação. A Web permite às pessoas armazenar vasta quantidade de informações para acesso público ou controlado. Com este crescimento gigantesco da quantidade de informação armazenada, fica cada vez mais evidente a necessidade de Sistemas de Recuperação de Informações (SRI) mais rápidos e mais eficientes nas buscas. Os SRI, no passado, eram utilizados em coleções de documentos menores, tais como bibliotecas e coleções particulares. Mas nos dias de hoje, eles podem e devem ser usados na Internet, por exemplo em Máquinas de Busca, que são *sites* especializados da Web usados para buscar informações disponíveis em outros *sites*.

Estes SRI direcionados para a Internet estão, cada vez mais, necessitando de novas técnicas que ajudem a torná-los mais poderosos. Uma destas técnicas que estão sendo usadas é a lógica Fuzzy. Há tempos tem sido mostrado que se pode usar a teoria dos conjuntos Fuzzy em recuperação de informação [ZDCK84].

Na lógica Fuzzy, ao contrário da lógica tradicional, os elementos possuem um certo grau de pertinência a determinados conjuntos (conjuntos Fuzzy). A partir deste grau de pertinência, é gerada uma função de pertinência (*membership function*) que é a base de todas as operações dos conjuntos Fuzzy.

O objetivo deste trabalho é estudar três diferentes aplicações da lógica Fuzzy na área de Recuperação de Informação, compará-las e propor uma máquina de busca eficiente para documentos na Web usando lógica Fuzzy.

Este trabalho está organizado em seis capítulos. O primeiro, já apresentado, esclarece o objetivo deste trabalho. O segundo capítulo, introduz os conceitos básicos de recuperação de informação, dando uma ênfase maior a

máquinas de busca. No terceiro capítulo, é apresentada a lógica Fuzzy, apresentando alguns conceitos e algumas características. No quarto capítulo, é feita uma integração entre recuperação de informação e lógica Fuzzy, com apresentação de três aplicações da lógica Fuzzy nos sistema de recuperação de informação. No quinto capítulo, é feita a proposta da máquina de busca que utiliza lógica Fuzzy. No sexto capítulo são apresentadas as conclusões e são sugeridos alguns trabalhos que poderão ser feitos no futuro.

CAPÍTULO 2

RECUPERAÇÃO DE INFORMAÇÃO

Recuperação de Informação (RI) é uma área responsável pela representação, armazenamento, organização e acesso a itens de informação. A representação e organização dos itens de informação fornecem ao usuário um fácil acesso à informação na qual ele está interessado. Infelizmente, a caracterização da necessidade de informação do usuário não é um problema simples. Este capítulo foi retirado quase que completamente de [BaRi99]. Portanto onde não houver outra referência, entende-se que seja [BaRi99].

Considere a seguinte descrição de um assunto que um usuário necessita na Web: procurar páginas que contenham informação dos times de futebol do Brasil, que participem do Campeonato Brasileiro da primeira divisão, informando suas colocações nos três últimos campeonatos, contendo ainda, cidade de origem com endereço completo, nome do seu presidente e o ano de fundação. Esta descrição não poderá ser usada diretamente para requisitar informação usando máquinas de busca na Web. Com isso, o usuário deverá primeiro traduzir estas informações em uma consulta que possa ser processada por uma máquina de busca. Esta tradução deve gerar um pequeno conjunto de palavras-chave que contenham a descrição da informação que o usuário necessita.

Recuperação de dados no contexto de um sistema de recuperação de informação consiste em determinar quais documentos de uma coleção contêm as palavras-chave que aparecem na consulta do usuário, e isto geralmente não é suficiente para satisfazer as suas necessidades de informação. O usuário de um sistema de recuperação de informação prefere que informações sejam recuperadas sobre determinado assunto (conceito), que contenham os dados que aparecem na consulta. O Sistema de Recuperação de Informação (SRI) deve

ordenar os documentos de acordo com a sua relevância (importância). Relevância é a palavra central de um sistema de recuperação de informação. É objetivo de um SRI buscar todos os documentos relevantes de uma consulta.

Nos últimos 20 anos, a área de RI tem crescido bastante. Nos dias de hoje, RI inclui estudos nas áreas de modelagem, classificação e categorização de documentos, interfaces amigáveis com usuários, visualização de dados, filtragem de informação, linguagens, dentre outros. A Web está tornando-se um repositório universal de conhecimento e cultura humana na qual tem permitido sem precedente compartilhando idéias e informação em escala nunca vista antes.

A recuperação eficaz de informações relevantes está diretamente ligada tanto com a tarefa do usuário, quanto com a visão lógica dos documentos.

2.1. A tarefa do usuário

Um usuário de um SRI tem que traduzir sua necessidade de informações em uma consulta e passá-la para uma linguagem fornecida pelo sistema. Isto implica em especificar um conjunto de palavras-chave que conduzam, de alguma forma, à semântica de sua necessidade. Neste caso, o usuário está buscando por informações úteis executando uma tarefa de recuperação.

Considere um exemplo em que um usuário esteja interessado em documentos que se referem a carros de corrida em geral. Neste caso, o usuário pode usar uma interface interativa para simplificar a procura em uma coleção de documentos que relatam carros de corrida. Ele pode procurar documentos sobre corrida de Fórmula 1 ou outro tipo de corrida. Enquanto ele estiver lendo sobre Fórmula 1, ele pode voltar sua atenção sobre outro tipo de corrida. Neste caso, diz-se que está navegando em uma coleção de documentos e não pesquisando. Pode-se fazer uma clara distinção entre as diferentes tarefas do usuário do sistema de recuperação pode ser engajado. Sua tarefa pode ser de dois tipos distintos: recuperação e navegação.

2.2. Visão lógica dos documentos

Por motivos históricos, documentos em uma coleção são geralmente representados por um conjunto de termos de indexação ou palavras-chave. Tais palavras podem ser extraídas automaticamente dos documentos ou podem ser especificados manualmente por algum especialista humano.

Computadores modernos, que possuem alto poder de armazenamento, estão tornando possível a representação de um documento pelo seu conjunto completo de palavras, é a chamada representação *full text*. Tal representação é a mais completa visão lógica de um documento, mas seu uso implica em custos computacionais altíssimos. Um conjunto menor de palavras selecionadas por especialistas humanos é a visão lógica mais concisa do documento, mas seu uso pode levar a uma recuperação de informação de baixa qualidade.

2.3. Passado, Presente e Futuro

Por aproximadamente 4000 anos o homem tem organizado informações para serem recuperadas e usadas posteriormente. Um exemplo típico é a tabela de conteúdo de um livro. Como o acervo de livros cresceu, uma estrutura teve de ser criada para acessar de forma mais rápida as informações armazenadas nos livros.

As bibliotecas estão entre as primeiras instituições a adotar o sistema de recuperação de informação. Sistemas para serem usados em bibliotecas eram inicialmente desenvolvidos por instituições acadêmicas e depois vendidas comercialmente. Na primeira geração, tais sistemas eram constituídos basicamente da automação de tecnologias anteriores. Na segunda geração, acrescentaram-se funções de busca por assuntos no cabeçalho, por palavras-chave e algumas consultas mais complexas. Na terceira geração, que é a que atualmente está sendo desenvolvida, o foco está em criar interfaces gráficas,

formas eletrônicas, características de hipertexto, e arquiteturas de sistemas abertos.

Considerando as máquinas de busca da Web atualmente, conclui-se que elas continuam usando índices similares aos usados por bibliotecários há séculos atrás. Porém, três mudanças drásticas ocorreram durante o avanço da tecnologia computacional e no crescimento da Web. Primeira, se tornou muito mais barato ter acesso a várias fontes de informação. Isto permite que seja realizado um número de pesquisa tão grande como nunca foi possível anteriormente. Segunda, o avanço em todos os tipos de comunicação digital produziu um acesso ainda maior às redes. Isto implica que as fontes de informações estão disponíveis, mesmo que localmente distantes e que o acesso pode ser realizado rapidamente. Terceira, a liberdade de divulgar qualquer tipo de informação que uma pessoa julgue útil, isto aumenta cada vez mais a popularidade da Web. Pela primeira vez na história, muitas pessoas têm acesso livre a uma enorme quantidade de publicações de médio e pequeno porte.

2.4. O Processo de Recuperação

Para descrever o processo de recuperação, usa-se uma arquitetura de *software* simples e genérico. Primeiramente, antes do processo de recuperação poder ser iniciado, é preciso definir o banco de dados de texto. Isto é usualmente feito por um administrador de banco de dados[*], que especifica: (a) os documentos para serem usados, (b) as operações para serem realizados no texto, e (c) o texto modelo.

O DBA cria um índice no banco de dados. Diferentes estruturas de índices podem ser usadas, mas o mais popular é o arquivo invertido, como mostrado na Figura 1, que ilustra de forma resumida, todo o processo de RI. Arquivo invertido será definido com mais detalhe na seção 2.5.5.

[*] (DBA – Database Administrator)

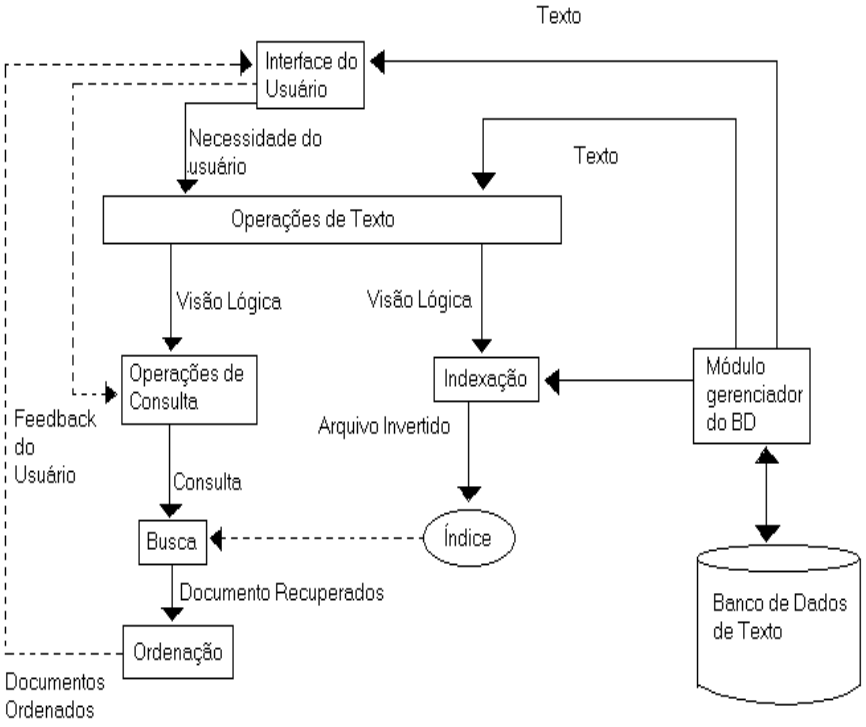

Figura 1: O processo de recuperação de informação

Dado que os documentos do banco de dados estão indexados, o processo de recuperação pode ser iniciado. O usuário primeiro especifica uma necessidade do usuário. Então, operações de consulta podem ser aplicadas antes que a consulta atual, que produz uma representação do sistema de necessidade do usuário, fosse gerada.

Antes de enviar ao usuário, os documento buscados são ordenados de acordo com sua relevância. O usuário então examina o conjunto de documentos ordenados na busca de informação útil.

2.5. Máquinas de Busca

Uma principal diferença entre SRI que foram desenvolvidos para alguma coleção padrão[*] e os para a Web é que, na Web, todas as consultas podem ser respondidas sem acesso aos textos dos documentos, somente os índices são avaliados. Diferentemente daquele que pode querer armazenar um ou outro para suprir uma cópia local de páginas da Web ou acessando páginas remotas direto na rede no tempo da consulta. Esta diferença tem impacto nos algoritmos de indexação e busca, e também linguagens de consultas podem ser avaliadas.

2.5.1. Arquitetura Centralizada

A maioria das máquinas de busca usa uma arquitetura centralizada *crawler-indexer*. *Crawlers* são programas (*software* agentes) que atravessam a Web enviando páginas novas ou atualizadas para o principal servidor onde eles estão indexadas. *Crawlers* são também chamados de robôs, aranhas, viajante ou caminhante. Um *crawler* atualmente não é executado em máquinas remotas, antes ele é executado em um sistema local e envia requerimentos para servidores Web remotos. O índice é usado em uma forma centralizada para responder consultas submetidas de diferentes lugares na Web.

2.5.2-Arquitetura Distribuída

Há muitas variantes da arquitetura *crawler-indexer*. Entre elas, a mais importante é a Harvest [CPDUM94]. Harvest usa uma arquitetura distribuída para reunir e distribuir dados, a qual é mais eficiente que a arquitetura *crawler*. A principal desvantagem é que Harvest requer a coordenação de vários servidores Web.

O Harvest distribuído aproxima vários endereços dos problemas da arquitetura *crawler-indexer*, tais como: (1) servidores Web recebem requisição

[*] Coleções de documentos preparados para testar a eficiência dos SRI.

de diferentes *crawlers* aumentando sua visão; (2) o tráfico na Web aumenta porque *crawlers* recuperam objetos inteiros, mas a maioria de seu conteúdo é descartada; e (3) informação é reunida independentemente por cada *crawler*, sem coordenação entre todas as máquinas de busca.

Para resolver estes problemas, Harvest introduz dois elementos principais: *gatherers* e *brokers*. Um *gatherer* coleta e extrai dados indexando informação de um ou mais servidores Web. As reuniões às vezes são definidas por um sistema e são periódicos. Um *broker* provém o mecanismo indexado e a interface de consulta para os dados reunidos. *Brokers* recuperam informação de um ou mais *gatherers* ou *brokers*, atualizando cada vez mais seus índices.

2.5.3. Interfaces com o usuário

Há dois aspectos importantes da interface com o usuário de máquinas de busca: a interface de consulta e a interface de resposta. A interface de consulta básica é uma caixa onde uma ou mais palavras podem ser digitadas. Além disso, um usuário pode esperar que uma dada seqüência de palavras represente a mesma consulta em todas as máquinas de busca, que não é verdade. Outro problema é que a visão lógica do texto não é conhecida, isto é, algumas máquinas de busca usam palavras de parada, alguns fazem estancamento, e alguns não são caso sensitivo.

A interface de resposta usualmente consiste de uma lista contém em cada tela aproximadamente as dez páginas com maior relevância ordenadas. Cada entrada nesta lista inclui alguma informação sobre o documento que ela representa. Tipicamente a informação inclui a URL, tamanho, a data de quando esta página foi indexada, e um par de linhas com seus conteúdos (título mais primeira linha ou escolhe o cabeçalho ou sentenças). Algumas máquinas de busca permitem ao usuário mudar o número de páginas retornadas na lista e a

quantidade de informação por página, mas na maioria dos casos isto é fixado ou limitado a umas poucas escolhas.

2.5.4. Ordenação (*Ranking*)

A maioria das máquinas de busca usa variações do modelo Booleano [BaRi99] para fazer a ordenação por relevância. Como com a busca, a ordenação tem sido feita sem acesso ao texto, somente ao índice. Não há muita informação publicada sobre os algoritmos de ordenação específicos usados pelas máquinas de busca correntes. É difícil comparar diferentes máquinas de busca dando suas diferenças, e melhoras contínuas. Mais importante, é quase impossível medir chamadas, como o número de páginas relevantes pode ser completamente grande de consultas simples.

Alguns dos novos algoritmos também usam informação *hiperlink*. Esta é uma diferença importante entre um SRI na Web e uma RI em bancos de dados normais. O número de *hiperlinks* naquele ponto para página provê uma medida de sua popularidade e sua qualidade. Além disso, muitos *links* em comum entre páginas e páginas referenciadas pela mesma página sempre indicam uma relação entre aquelas páginas.

2.5.5-Índices

A maioria dos índices usa variantes do arquivo invertido. Um arquivo invertido é uma lista de palavras (termos) ordenadas chamadas de vocabulário, cada uma contendo um conjunto de ponteiros para as páginas onde ela ocorre. Algumas máquinas de busca usam a eliminação de palavras com pouca representação semântica (*stopwords*) para reduzir o tamanho dos índices. Além disso, é importante lembrar que é a visão lógica do texto que é indexada. Operações de normalização podem incluir remoção de pontuação e múltiplos espaços para cada espaço entre cada palavra. Para dar uma idéia ao usuário sobre cada

documento recuperado, o índice é completado com uma pequena descrição de cada página da Web (data de criação, tamanho, o título e as primeiras linhas ou um pequeno cabeçalho são típicos).

O arquivo invertido pode também apontar para a ocorrência atual de uma palavra dentro de um documento (*full inversion*). Como sempre, isto é de grande valor no espaço da Web, porque cada apontador tem que especificar uma página e uma posição dentro da página (número de palavras que podem ser usadas além dos bytes atuais). Por outro lado tendo a posição das palavras na página, pode-se responder a frases de busca ou consultas aproximadas pela procura de palavras que estão perto de cada outra na página. Algumas máquinas de busca estão provendo buscas com frases, mas a implementação atual ainda não é a mais apropriada.

2.5.6 Modelo Booleano

Em Recuperação de Informação existem três modelos clássicos: Booleano, Vetorial [BaRi99] e Probabilístico [BaRi99]. Destes três modelos clássicos, o modelo Booleano é o mais utilizado, sendo que a maioria dos Sistemas de Recuperação de Informações, inclusive os já tradicionais *sites* de busca, usa este modelo em sua estrutura. Este modelo usa as tradicionais expressões Booleanas (AND, OR e NOT) para formular a consulta do usuário.

CAPÍTULO 3
LÓGICA FUZZY

3.1 Introdução à lógica

Lógica é o estudo dos métodos e princípios do raciocínio em todas as suas formas possíveis [KlYu95].

Aristóteles, filósofo grego (384 - 322 a.C.), foi o fundador da ciência da lógica, e estabeleceu um conjunto de regras rígidas para que conclusões pudessem ser aceitas logicamente válidas. O emprego da lógica de Aristóteles levava a uma linha de raciocínio lógico baseado em premissas e conclusões. Desde então, a lógica Ocidental, assim chamada, tem sido binária (booleana), isto é, uma declaração é falsa ou verdadeira, não podendo ser ao mesmo tempo parcialmente verdadeira e parcialmente falsa. Os conjuntos clássicos possuem fronteiras bem definidas que diferenciam com precisão os membros dos não-membros do conjunto [ARBE94].

3.2 Introdução à Lógica Fuzzy

O conceito de lógica Fuzzy, também chamada de lógica Difusa, foi introduzido, em 1965, por Lotfi A. Zadeh [ZADE65].

Ele teve duas idéias, uma considerada brilhante e outra considerada no mínimo infeliz. A idéia brilhante foi uma nova lógica e uma nova teoria dos conjuntos onde não precisamos nos contentar com apenas duas opções (verdadeiro ou falso, pertence ou não pertence), mas com um grau infinito que varia entre esses duas. Assim, podemos ter algo que é 50% falso ou pertence ao conjunto apenas 30%. A idéia considerada infeliz foi chamar essa nova teoria de Fuzzy, que significa “nebulosa”, “difusa”, "de forma indistinta", "de imagem confusa", "cabeluda", "gasta (como uma roupa fica gasta)" e "peluda (como animais)" [XEXE98].

Essa escolha fez com que houvesse pouco interesse na lógica nebulosa, ou difusa, como podemos dizer no Brasil, principalmente nos EUA, onde o termo era carregado de significados indesejados em um trabalho sério. No Japão, porém, as soluções propostas por Zadeh começaram a ser utilizadas com sucesso, o que chamou a atenção de todos (principalmente EUA e alguns países da Europa) para o que hoje é conhecido como uma das mais interessantes teorias para implementar sistemas de controle e sistemas de inteligência artificial.

Zadeh observou que, naquela época, os recursos tecnológicos disponíveis eram incapazes de automatizar as atividades que compreendessem situações ambíguas, não passíveis de processamento através da lógica computacional fundamentada na lógica booleana.

A teoria Fuzzy é utilizada para representar modelos de raciocínio impreciso, que possuem um papel essencial na notável habilidade humana, para tomar decisões racionais em ambientes de incertezas e imprecisões [BXR02]. A lógica Fuzzy pode ser definida como sendo uma ferramenta capaz de capturar informações vagas, em geral descritas em uma linguagem natural e convertê-las para um formato numérico, de fácil manipulação pelos computadores de hoje em dia, ou ainda, como a lógica que suporta os modos de raciocínio que são aproximados, ao invés de exatos, como estamos naturalmente acostumados a trabalhar [TRY02].

A lógica Fuzzy é basicamente uma lógica multivalorada que permite valores intermediários para ser definidos entre avaliações convencionais como sim/não, verdadeiro/falso, branco/preto etc. Noções como muito quente e meio frio podem ser formuladas matematicamente, formando os conjuntos Fuzzy e processadas pelos computadores [BNW96].

Com o uso da lógica Fuzzy, muitas vantagens podem ser notadas. Algumas destas vantagens estão mostradas a seguir:

- Requer poucas regras, valores e decisões;
- Mais variáveis observáveis podem ser valoradas;
- O uso de variáveis lingüísticas ("muito grande", "pouco frio", "mais ou menos jovem") é mais aproximado do pensamento humano;
- Simplifica a solução de problemas;
- Proporciona um rápido protótipo dos sistemas;
- Simplifica a aquisição da base do conhecimento.

3.3 Conjuntos Fuzzy

Um conjunto nebuloso é uma extensão do conceito de conjunto *crisp* onde a imagem da função característica deixa de ser o conjunto {0,1} e passa a ser o intervalo [0,1], com 0 representando totalmente falso e 1 representando totalmente verdadeiro. Passamos a chamar essa função de função de pertinência *(membership function)* [ZADE65]. Na figura 2 temos um exemplo de pertinência nos conjuntos Fuzzy.

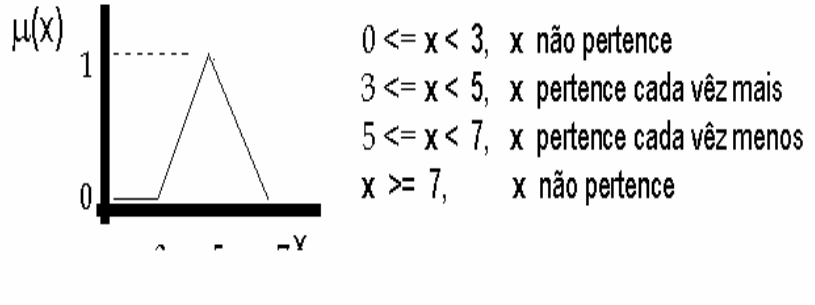

Figura 2: Pertinência em conjuntos Fuzzy

Um conjunto nebuloso *A* definido no universo de discurso *X* é caracterizado por uma função de pertinência μ_A, a qual mapeia os elementos de *X* para o intervalo [0 , 1] [KlYu95]:

$$\mu A : X \Rightarrow [0 , 1]$$

Em uma outra representação, a função é denotada somente por *A* e tem, claro, a mesma forma:

$$A : X \Rightarrow [0 , 1]$$

Cada conjunto Fuzzy é completamente e unicamente definido por uma função de pertinência particular. Na Figura 3 é mostrado um exemplo, onde o grau de altura representa a pertinência de cada pessoa ao conjunto das pessoas altas:

Pessoa	**Altura**	**Grau de altura**
Mônica	1,40m	0,3
Eduardo	1,70m	0,8
Carlos	1,90	1,0
João	2,10	1,0

Figura 3: Exemplo da pertinência em um conjunto Fuzzy. Uma pessoa com altura entre 1,50m e 2,00m, possui seu grau de altura (pertinência) variando entre 0,0 e 1,0.

Algumas características dos conjuntos Fuzzy:

- A lógica difusa está baseada em palavras e não em números, ou seja, os valores verdades são expressos lingüisticamente. Por exemplo: quente, muito frio, verdade, longe, perto, rápido, vagaroso, médio, etc.
- Possui vários modificadores de predicado como por exemplo: muito, mais ou menos, pouco, bastante, médio, etc.

- Possui também um amplo conjunto de quantificadores, como por exemplo : poucos, vários, em torno de, usualmente.
- Faz uso das probabilidades lingüísticas, como por exemplo : provável, improvável, que são interpretados como números Fuzzy e manipulados pela sua aritmética.

Manuseia todos os valores entre 0 e 1, tomando estes, como um limite apenas.

Qualquer representação adequada de um conjunto Fuzzy envolve o entendimento básico de cinco diferentes símbolos conceituais, relacionados entre si [BXR02]:

- **Conjunto de elementos** , como, por exemplo, um "homem" em "homens" ou um "item" em "estoque".
- **Variável lingüística *V*** em um conjunto de variáveis lingüísticas, que é um rótulo para um atributo dos elementos , como "altura de homem" ou o "nível de estoque" de uma empresa.
- **Termo lingüístico *A*** de uma *variável lingüística*, correspondendo a um adjetivo ou a um advérbio, em um conjunto de termos lingüísticos, como "homem alto" associado com a "altura do homem" ou "estoque baixo", relacionado com possíveis "níveis de estoque" de uma empresa.
- **Intervalo numérico mensurável *X***, conhecido como o conjunto referencial para um atributo particular ***V***, de um conjunto de elementos , como, por exemplo, "[0, 3] metros" para "altura de homem", ou "[250, 750] unidades" para "nível de estoque".
- **Atribuição numérica subjetiva A(), ou valor de pertinência**, que é o grau com que um elemento pertence ao conjunto de elementos, rotulados por uma variável lingüística ***V***, e identificados pelo termo lingüístico **A**. Por exemplo, o valor de pertinência dado a um "homem" em um grupo de homens por um observador, que usa o termo lingüístico

"alto", segundo sua visão de "altura" para homens, ou o valor de pertinência atribuído por um gerente para "estoque", através do adjetivo "baixo", englobando todos os níveis de estoque sob o seu gerenciamento.

Os conjuntos Fuzzy, assim como os outros, possuem algumas operações. As operações de VAZIO, IGUALDADE, COMPLEMENTO (NOT), CONTER, UNIÃO (OR), e INTERSEÇÃO (AND) serão definidas a seguir [BRU85]:

Sendo X um conjunto de objetos, um conjunto Fuzzy A em X é caracterizado pela função de pertinência $\mu_A(x)$ que mapeia cada ponto em x para o intervalo real [0.0, 1.0]. Quando μ_A aproxima de 1.0, o "grau de pertinência" de x em A aumenta. Dessa forma:

- A é VAZIO se para todo x, $\mu_A(x) = 0.0$.
- $A = B$ se para todo x: $\mu_A(x) = \mu_B(x)$ [ou, $\mu_A = \mu_B$].
- $\mu_A' = 1 - \mu_A$.
- A ESTÁ CONTIDO em B se $\mu_A \leq \mu_B$.
- $C = A$ UNIÃO B, onde: $\mu_A(x) = \text{MAX}(\mu_A(x), \mu_B(x))$.
- $C = A$ INTERSEÇÃO B onde: $\mu_C(x) = \text{MIN}(\mu_A(x), \mu_B(x))$.

CAPÍTULO 4

INTEGRAÇÃO ENTRE RECUPERAÇÃO DE INFORMAÇÃO E LÓGICA FUZZY

Há tempos tem sido mostrado que se pode usar a teoria dos conjuntos Fuzzy em recuperação de informação. Um usuário de um sistema de informação, interessado em certos tópicos, freqüentemente processa sua consulta para um sistema como uma seqüência lógica de termos que podem ser igualada com documentos do sistema.

A generalização da idéia Booleana de que um documento contém totalmente um termo ou não o contém todo, para uma idéia mais realística que explica que um documento pode incluir um termo a um certo grau é perfeitamente descrita em termos de conjuntos Fuzzy. Neste capítulo serão apresentadas três diferentes aplicações da lógica Fuzzy em recuperação de informação.

Na primeira aplicação, como a maioria dos procedimentos não calcula estimativa de tempo, nem estimativa do armazenamento necessário para recuperar a resposta da consulta do usuário, Radecki [RADE77] descreveu o método de redução usando conjuntos Fuzzy *lambda-level*. A definição de *lambda-level* será feita na seção 4.1. Mas neste caso há dependência do valor de *lambda* usado.

Na segunda aplicação, será tratado o processo de fazer julgamento de relevância. Pega-se a visão em que usuários tomam decisão de relevância pela quebra do problema de relevância em pequenas partes [KMSP88].

- Primeiro o julgamento é feito com porções menores da consulta.
- Depois aqueles julgamentos são vistos juntos e projetado um julgamento da consulta completa. Aqui, um julgamento com o conceito de alto nível da consulta toda é referenciado como um julgamento “complexo” enquanto julgamentos feitos com um conceito de nível menor em uma

consulta são chamadas de julgamento "atômicos". São consideradas palavras individuais para representar conceito de nível menor.

- Cada documento adquire um valor com respeito a cada termo de consulta, e então os valores individuais são combinados num valor simples para representar a recuperação do valor de *status* do documento com respeito à consulta complexa. O ponto de foco é o mecanismo de inferência usado para derivar o julgamento complexo do atômico.

Examina-se o quão bem o processo de inferência é modelado pela teoria probabilística e pela teoria do conjunto Fuzzy. O modelo probabilístico discute qual o tamanho do termo pode ser combinado probabilisticamente. A lógica Fuzzy supre um quadro de trabalho lógico consistente de avaliação de documentos. Um sistema de recuperação de documentos pode avaliar os documentos da mesma forma que o usuário pode.

Na terceira aplicação, é proposto um sistema de recuperação de documentos Fuzzy usando uma matriz de conexão de palavras-chave para representar similaridades entre palavras-chave. Pelo uso da matriz de conexão de palavras-chave, documentos são classificados de acordo com suas relevâncias à consulta do usuário. Também é proposto um método de aprendizagem para modificar o valor de pertinência e reduzir a diferença entre o valor de pertinência inicialmente apontado usando informação estatística, e avaliação do usuário [OMK89].

Há, contudo, dois problemas com o último método: Primeiro, as consultas complexas foram restritas a OR lógico; segundo, o método não pode aceitar julgamento ambíguo de usuário no processo de aprendizagem. Este método é estendido para resolver estes dois problemas. No novo método de consultas complexas compostas de palavras-chave com AND, OR e/ou NOT são processadas, e o método de aprendizagem tem sido modificado para permitir julgamento Fuzzy tão bem quanto consultas complexas.

4.1 Uma nova aproximação a sistemas de recuperação de informação usando expressões Fuzzy

Este é a primeira aplicação de lógica Fuzzy em sistemas de recuperação de informação [ZDCK84].

4.1.1 Noções Básicas

Definição 1. Por um Sistema de Recuperação de Informação, *I*, podemos definir uma quádrupla

$$I = \{D,Q,T,Y\}$$

Onde:

- *D* é um conjunto finito de documentos avaliáveis;
- *Q* é um conjunto finito de consultas que podem ser processadas por um sistema;
- *T* é um conjunto finito de termos;
- *Y* é uma projeção de *Q* a *D*, isto é, *Y* associa cada consulta *q* que pertence a *Q* um subconjunto *Y(q)* do conjunto de documentos *D*.

O Sistema de recuperação de documento pode agora ser descrito pelo significado de uma relação Fuzzy binária *F*:

$$F = \{[(d,t),\mu_{(d,t)}] \mid d \in D, t \in T\}$$

Com a função de pertinência

$$\mu : D \text{ x } T \Rightarrow [0,1],$$

$$(d,t) \Rightarrow \mu(d,t), \quad \text{para todo } (d,t) \text{ pertence a } DxT,$$

para que à medida que um documento *d* proceda com um termo *t*. Escrevemos *F*(*d*,*t*) em vez de μ(*d*,*t*).

A relação *F* descreve o quanto um documento é relevante a um termo, em vez de indicar a importância de um termo nesse documento.

Definição 2. Radecki [RADE79] define a linguagem *L* do sistema de recuperação de informação como um alfabeto, consistindo dos seguintes símbolos:

- Um conjunto *T* de termos básicos
- Conectivos lógicos: negação representado por ’ ; conjunção representado por · ; e disjunção representado por +.
- Sinal de igualdade: =.
- Parênteses: ( ), [ ], { }, ...,

Junto com o conjunto *T** de termos complexos, definidos recursivamente pelas seguintes regras sintáticas:

- Para todo *t* pertencente a *T* : *t* pertence a *T**
- Para todo *t* pertencentes a *T** : *t*’ pertence a *T**
- Para todo t_1, t_2 pertencentes a *T** : $t_1 \cdot t_2$ pertencem a *T**
- Para todo t_1, t_2 pertencentes a *T** : $t_1 + t_2$ pertencem a *T**
- Os elementos de *T** são somente aqueles formados pela aplicação das regras 1-4.

Definição 3. Para todo termo básico *t* pertencente a *T* pode definir o significado de *t* como um subconjunto Fuzzy *Mt* de *D*:

$$Mt = \{[d, Mt_{(d)}] \mid d \text{ pertence a } D \text{ e } Mt_{(d)} = F(d,t)\}.$$

Como a resposta tem que ser dada em tempo aceitável, a otimização do processo de busca pode ser aumentada pela escolha do valor ótimo de *lambda*, com respeito ao tempo e à qualidade do processo de recuperação, e pela operação no significado *lambda-level* dos termos.

Definição 4. A relação *lambda-level (F*λ) é definida como

$$F_\lambda = \{[(d,t), F_\lambda(d,t)] \mid (d,t) \in DxT\}$$

Onde

$$F_{\lambda}\ (\mathrm{d,t}) = \begin{cases} F(d,t), se F(d,t) \geq \lambda \\ 0, se F(d,t) < \lambda \end{cases}$$

Definição 5. O significado *lambda-level* $Mt_{(\lambda)}$ de um termo básico *t* pertencente a *T* é definido como o conjunto Fuzzy em *D*,

$$Mt_{(\lambda)} = \{[d, Mt_{(\lambda)(d)}] \mid d \in D, Mt_{(\lambda)(d)} = F_{\lambda(d,t)}\}$$

O significado *lambda-level* $Mt'_{(\lambda)}$ do termo complexo *t'*, onde *t* pertence a *T**, é definido como

$$Mt'_{(\lambda)} = ([d, Mt'_{(\lambda)(d)}] \mid d \in D\}$$

Onde

$$Mt'_{(\lambda)(d)} = \begin{cases} 1 - Mt_{(\lambda)(d)}, se 1 - Mt_{(\lambda)(d)} \geq \lambda, \\ 0, se 1 - Mt_{(\lambda)(d)} \leq \lambda. \end{cases}$$

O significado *lambda-level* $M(t_1 \cdot t_2)_{(\lambda)}$ do termo complexo $t_1 \cdot t_2$ onde t_1, t_2 pertencem a *T**, é definido por

$$M(t_1 \cdot t_2)_{(\lambda)} = \{[d, M_{(t1 \cdot t2)(\lambda)(d)}] \mid d \in D\}$$

Onde

$$M(t_1 \cdot t_2)_{(\lambda)(d)} = \min[Mt_{1(\lambda)(d),} Mt_{2(\lambda)(d)}]$$

O significado *lambda-level* $M(t_1 + t_2)_{(\lambda)}$ do termo complexo $t_1 + t_2$ onde t_1, t_2 pertencem a *T**, é definido por

$$M(t_1 + t_2)_{(\lambda)} = \{[d, M(t_1 + t_2)_{(\lambda)(d)}] \mid d \in D\}$$

Onde

$$M(t_1 + t_2)_{(\lambda)(d)} = \max[Mt_{1(\lambda)(d)}, Mt_{2(\lambda)(d)}].$$

Usando a definição clássica de Zadeh de interseção e união, obtém-se:

$$M(t_1 \cdot t_2)_{(\lambda)} = Mt_{1(\lambda)} \cap Mt_{2(\lambda)} \qquad M(t_1 + t_2)_{(\lambda)} = Mt_{1(\lambda)} \cup Mt_{2(\lambda)}$$

Mas $Mt'_{(\lambda)} \neq coMt_{(\lambda)}$ onde $coMt_{(\lambda)}$ denota o complemento do conjunto Fuzzy $Mt_{(\lambda)}$.

A última desigualdade é responsável pelos problemas levantados quando operado no significado *lambda-level*. De, qual é mais útil, o significado *lambda-level* $Mt'_{(\lambda)}$ do termo complexo *t'* ou o complemento do subconjunto Fuzzy de $Mt_{(\lambda)}$?

$$coMt_{(\lambda)} = \{[d,b] \mid d \in D\} \text{ onde } b = \begin{cases} 1 - Mt_{(\lambda)(d)}, se Mt_{(\lambda)(d)} \geq \lambda, \\ 1, se Mt_{(\lambda)(d)} < \lambda. \end{cases}$$

e

$$Mt'_{(\lambda)} = \{[d,b] \mid d \in D\} \text{ onde } b = \begin{cases} 1 - Mt_{(\lambda)(d)}, se 1 - Mt_{(\lambda)(d)} \geq \lambda, \\ 0, se Mt_{(\lambda)(d)} < \lambda. \end{cases}$$

Pode também ser notado que a definição original de Radecki é também insatisfatória, porque no caso $Mt'_{(\lambda)}$ torna-se

$$Mt'_{(\lambda)} = \{[d,b] \mid d \in D\}$$

Onde

$$b = \begin{cases} 1 - Mt_{(\lambda)(d)}, se Mt_{(\lambda)(d)} \geq \lambda, 1 - Mt_{(\lambda)(d)} \geq \lambda, \\ 0, se Mt_{(\lambda)(d)} \geq \lambda, 1 - Mt_{(\lambda)(d)} < \lambda, \\ 1. \end{cases}$$

4.1.2 Descrição de um modelo corrente de recuperação de documento

Será descrito um algoritmo, que aloca documentos para consultas particulares. Neste sistema, cada consulta *q* é representada por um termo complexo *t* que pertence a *T**, devolvendo o conteúdo da consulta no melhor caminho.

Algoritmo. O processo de busca que é seguido quando uma consulta *q* pertencente a *Q* é processada pelo sistema pode ser descrita pelos seguintes passos:

1. Formar, de acordo com as regras sintáticas da linguagem *L*, um termo complexo *t* pertencente a *T** que representa a consulta *q*.
2. Determinar o significado *lambda-level* do termo complexo *t*,

$$Mt_{(\lambda)} = ([d, Mt_{(\lambda)(d)}] \mid d \in D\}$$

3. Ordenar os documentos de $Mt_{(\lambda)}$ em ordem decrescente de pertinência.

A resposta do sistema *Y(q)* desta lista ordenada de documentos, junto com seus graus de pertinência. Documentos pertencendo a $Mt_{(\lambda)}$ de ordem zero não são mencionados em *Y(q).*

Exemplo. Considere o conjunto de documentos *D = {d1, d2, d3, d4, d5, d6, d7}* e o conjunto de termos *T = {t1, t2, t3, t4, t5}*. Supondo que o valor ótimo de *lambda* λ deste sistema de recuperação de documentos é 0.2. A Tabela 1 mostra a relação *F* que descreve os documentos *d* pertencentes a *D*.

Tabela 1: Relação que descreve os documentos $d \in D$.

	t1	t2	t3	t4	t5
d1	0.6	0.9	0.8	0.14	0.9
d2	0.9	0.15	0.4	0.4	0.8
d3	0.2	0.4	0.9	0.9	1.0
d4	0.1	0.3	1.0	1.0	0.1
d5	0.9	0.4	0.2	0.0	0.9
d6	0.4	0.2	0.8	0.4	0.1
d7	0.3	0.1	0.3	0.3	1.0

A interseção da linha *i* com a coluna *j* é $F(d_i, t_j)$, indicando a posição a qual o documento d_i procede com o termo t_j. A relação suprida é a relação 0.2-*level* $F_{(0.2)}$, mostrada na Tabela 2.

Tabela 2: Relação 0.2-level $F_{(0.2)}$.

	t1	t2	t3	t4	t5
d1	0.6	0.9	0.8	0.0	0.9
d2	0.9	0.0	0.4	0.4	0.8
d3	0.2	0.4	0.9	0.9	1.0
d4	0.0	0.3	1.0	1.0	0.0
d5	0.9	0.4	0.2	0.0	0.9
d6	0.4	0.2	0.8	0.4	0.0
d7	0.3	0.0	0.3	0.3	1.0

Suponha agora que é dada uma consulta *q* pertencente a *Q* e é representada pelo termo complexo *t* pertencente a *T**, dada por

$$t = (t_1 \cdot (t_3 + t_5)') +) t_1 ' \cdot (t_3 + t_5)).$$

Sucessivamente, obtém-se:

$M_{t_1(0.2)} = \{(d_1, 0.6), (d_2, 0.9), (d_3, 0.2), (d_4, 0.0), (d_5, 0.9), (d_6, 0.4), (d_7, 0.3)\}$,

$M_{t_3(0.2)}$ ={(d_1,0.8), (d_2,0.4), (d_3,0.9), (d_4,1.0), (d_5,0.2), (d_6,0.8), (d_7,0.3)},

$M_{t_5(0.2)}$ ={(d_1,0.9), (d_2,0.8), (d_3,1.0), (d_4,0.0), (d_5,0.9), (d_6,0.0), (d_7,1.0)},

$M_{(t_3+t_5)(0.2)}$ ={(d_1,0.9), (d_2,0.8), (d_3,1.0), (d_4,1.0), (d_5,0.9), (d_6,0.8), (d_7,1.0)},

$M_{(t_3+t_5)'(0.2)}$ ={(d_1,0.0), (d_2,0.2), (d_3,0.0), (d_4,0.0), (d_5,0.0), (d_6,0.2), (d_7,0.0)},

$M_{(t_1\cdot(t_3+t_5)'(0.2)}$ = {(d_1,0.0), (d_2,0.2), (d_3,0.0), (d_4,0.0), (d_5,0.0), (d_6,0.2), (d_7,0.0)},

$M_{t_1'(0.2)}$ ={(d_1,0.4), (d_2,0.0), (d_3,0.8), (d_4,1.0), (d_5,0.0), (d_6,0.6), (d_7,0.7)},

$M_{(t_1'\cdot(t_3+t_5))(0.2)}$ = {(d_1,0.4), (d_2,0.0), (d_3,0.8), (d_4,1.0), (d_5,0.0), (d_6,0.6), (d_7,0.7)},

Até que:

$M_{t(0.2)}$ = {(d_1,0.4), (d_2,0.2), (d_3,0.8), (d_4,1.0), (d_5,0.0), (d_6,0.6), (d_7,0.7)},

A resposta do sistema Y(q) torna-se:

Documento d_4 satisfaz a consulta com 1.0
Documento d_3 satisfaz a consulta com 0.8
Documento d_7 satisfaz a consulta com 0.7
Documento d_6 satisfaz a consulta com 0.6
Documento d_1 satisfaz a consulta com 0.4
Documento d_2 satisfaz a consulta com 0.2

Observação. Várias consultas estão sendo feitas com diferentes valores de *lambda*. Na maioria das consultas, apareceram que o grau a qual o documento

satisfez uma consulta dependia do valor de *lambda* usado. Se o valor de *lambda* usado no exemplo anterior fosse 0.4, a resposta do sistema *Y(q)* podia ser:

Documento d_4 satisfaz a consulta com 1.0
Documento d_3 satisfaz a consulta com 1.0
Documento d_7 satisfaz a consulta com 1.0
Documento d_6 satisfaz a consulta com 0.6
Documento d_1 satisfaz a consulta com 0.4

Observa que o valor de d_2 saiu da lista porque ele é menor do que o valor de *lambda*. O quanto um documento é bom para uma consulta depende do valor de *lambda*.

Se deseja-se aqueles documentos que satisfazem muito a consulta, deve-se usar o valor de *lambda* maior (por exemplo λ=0.6). Se a resposta do sistema não contém um número satisfatório de documentos – do ponto de vista do usuário – ele pode conseguir mais documentos usando valores menores para *lambda*.

4.2 Modelo Fuzzy versus Modelo Probabilístico para julgamento de relevância do usuário

Um aspecto importante na área de recuperação de informação é o processo de fazer julgamento de relevância. Nesta aplicação, há a visão que, dado um documento e uma consulta, usuários primeiro julgam o documento com um conceito individual na consulta e então usam algum processo de inferência daquela decisão atômica para um julgamento complexo de toda consulta. Os modelos probabilístico e Fuzzy são usados como dois pontos de referência que para analisar a decisão do usuário em fazer julgamento complexos [KMSP88].

4.2.1 Aproximação

A aproximação envolve a aplicação de dois modelos, o modelo probabilístico e o modelo de conjuntos Fuzzy, para predizer julgamentos “complexos”, dado o julgamento “atômico” do usuário. As predições são então comparadas àquelas feitas diretamente pelo usuário. O argumento é o seguinte: pode-se ser mostrado que o modelo probabilístico melhor se aproxima à decisão do usuário então uma possível conclusão é que o usuário tem um sentimento intuitivo da probabilidade atômica de relevância e manipula-os dentro de uma junta de probabilidades de julgamentos complexos. Se o modelo Fuzzy é melhor na aproximação do julgamento humano, então pode ser concluído que o processo de decisão envolve regras de classe de pertinência.

Ambos os modelos começam pela designação numérica de peso para termos individuais em documentos (e possivelmente em consultas). Os modelos variam em suas interpretações de seus pesos. Estes valores significantes são usados para produzir a ordenação dos documentos por relevância. O modelo probabilístico é baseado em uma teoria elegante, enquanto o processo do modelo Fuzzy encara alguma censura.

4.2.2 O modelo probabilístico

O modelo probabilístico não considera operadores Booleano. Croft [CROF86] considera o operador AND como um indicador de importante dependência especificado do usuário. Em suas estratégias de ordenação eles primeiro calculam valores de cada documento sobre o termo independente assumido.

Depois esses valores são corrigidos pelo uso da informação provida pelas dependências especificadas.

O modelo probabilístico é baseado no princípio da ordenação da probabilidade que tenta ordenar os documentos, em resposta a consulta, em ordem decrescente de probabilidade de relevância. Isto é, se X, um vetor de

tamanho binário de termos indexados, é a descrição de um documento, então este ordenação do documento é determinado pela probabilidade condicional *P*(relevância | *X*). Usando o teorema de Bayes [BaRi99], que é convencionalmente usado para prover uma ordenação equivalente é a probabilidade estimada de *X* representando um documento, dado que o documento é relevante, isto é, *P*(*X* | relevância). Se $X=\{x_1, x_2, x_3, ..., x_n\}$, então a seguir há condições de independência de termos:

$$P(X \mid \text{relevância}) = P(x_1 \mid \text{relevância}) * P(x_2 \mid \text{relevância}) * P(x_3 \mid \text{relevância}) * ... * P(x_n \mid \text{relevância}) \quad (1)$$

Aqui cada $P(x_i \mid \text{relevância})$ é normalmente estimado usando a distribuição estatística do termo x_i no conjunto de documentos relevantes e não relevantes (no que diz respeito a uma consulta). A expressão (1) é usada para estimar a significância de cada documento com respeito a uma consulta.

Consultas são também compostas de termos, mas com opções acrescentadas de incluir conectivos Booleanos. As probabilidades de interesse aqui são *P*(relevância | *Q*) e *P*(*Q* | relevância), onde *Q* é a descrição de uma consulta. A primeira representa a probabilidade de relevância (de um documento) a uma consulta particular, enquanto a segunda representa a probabilidade de *Q* representando ou especificando a consulta dado que o documento é relevante.

P(relevância | *Q*) pode ser usado para ordenar consultas. O teorema de Bayes indica que uma ordenação equivalente pode ser obtida pelo uso de *P*(*Q* | relevância). Se *Q* é o conjunto de termos da consulta $\{q_1, q_2, q_3, ..., q_m\}$, então abaixo a suposição de termo independente da consulta, nós temos:

P(Q | relevância) = $P(q_1$ | relevância) * $P(q_2$ | relevância)* $P(q_3$ | relevância) ...*$P(q_m$ | relevância) (2).

Se Q é uma combinação Booleana de termos de consulta então nós usamos a seguinte expansão: seja q_1 e q_2 dois termos de consulta e

$P(q_1$ | relevância) = w_1 e $P(q_2$ | relevância) = w_2

então

$P(q_1$ AND q_2 | relevância) = $w_1 * w_2$ (3)

$P(q_1$ OR q_2 | relevância) = $w_1 + w_2 - w_1 w_2$ (4)

e

P(NOT q_1 | relevância) = 1 - w_1 (5)

Estas expressões podem ser usadas para predizer o julgamento complexo do julgamento atômico. Pode-se ainda combinar os conectivos para formar uma consulta:

$$P((q_1 \text{ AND } (q_2 \text{ OR } q_3)) \mid \text{relevância}) = (w_1)(w_2 + w_3 - w_2 w_3).$$

4.2.3 Modelo Fuzzy

Este modelo pega uma aproximação da teoria de conjuntos Fuzzy para a relação entre documentos e termos. O peso numérico dado a termo de consulta e par de documentos representa o grau de pertinência daquele documento ao conjunto de documentos que estão sobre aquele termo. Dado o assunto *Y*, o modelo discute pertinência de um documento ao conjunto de documentos sobre *Y*. Se o tópico *Y* é 'atômico', isto é, representado por um termo apenas então o valor de pertinência é simples. De qualquer modo, consultas ou tópicos podem ser

representados pela combinação Booleana de termos. O modelo pode ser usado para determinar pertinência de um documento ao conjunto de documentos sobre tais tópicos complexos. O procedimento utiliza o valor de pertinência do documento aos conjuntos correspondendo aos sub-tópicos da consulta.

Para ilustrar, seja $fq_{1(d)}$ e $fq_{2(d)}$ o valor Fuzzy apontado aos termos da consulta q_1 e q_2, respectivamente, em associação com o documento *d*. Agora, o valor Fuzzy associado com diferentes combinações de q_1 e q_2 são como as seguintes:

$$f(q_1 \text{ AND } q_2)(\text{d}) = \min (fq_1 (\text{d}), fq_2 (\text{d})) \quad (6)$$

$$f(q_1 \text{ OR } q_2)(\text{d}) = \max (fq_1 (\text{d}), fq_2 (\text{d})) \quad (7)$$

$$f(\text{NOT } q_1)(\text{d}) = 1 - fq_1 (\text{d})$$

Estas expressões podem ser usadas para predizer julgamentos complexos dos atômicos. Aqui também pode haver combinação de conectivos:

$$F(q_1 \text{ AND } (q_2 \text{ OR } q_3)(\text{d}) = \min(fq_1 (\text{d}), \max(fq_2 (\text{d}), fq_3 (\text{d}))).$$

Há diferentes caminhos de determinar o valor Fuzzy individual: fq_i (d)s. O fq_i (d)s são providos pelo usuário na forma de julgamento numérico feito por cada documento contra os termos da consulta individual.

4.2.4 Desenvolvimento de hipótese e experimento

A hipótese nula é que não há diferença entre o modelo Fuzzy e o modelo probabilístico em determinação a uma percepção do usuário de relevância do documento. Estatisticamente testado:

$$H_0 : r(\text{UF}) - r(\text{UP}) = 0$$
$$H_1 : r(\text{UF}) - r(\text{UP}) \neq 0$$

Onde *r*(UF) é a *Pearson Product Moment Correlation* entre a percepção de relevância do usuário e a predição de relevância feita pelo modelo Fuzzy e *r*(UP) é a correlação entre a percepção do usuário e a predição de relevância feita pelo modelo probabilístico.

Para começar, as buscas no banco de dados on-line foi usado para recuperaram dois conjuntos (10 documentos em cada) de títulos e *abstracts* de documentos. Os dois conjuntos eram de diferentes assuntos. Um grupo de quatro termos importantes de cada consulta foi selecionado do conjunto de documentos correspondentes. Estes termos formaram as consultas de termo único. Estes termos únicos eram também combinados para formar consultas Booleanas específicas de cada conjunto. Há quatro conjuntos complexos de cada conjunto. A Figura 4 lista todas as consultas usadas. Consultas de 1 a 8 na figura pertencem ao primeiro conjunto, enquanto as consultas 9 a 16 pertencem ao segundo. Como a figura mostra, houve duas consultas que usaram somente o operador OR, cinco que usaram somente AND enquanto uma, a consulta 6, usou ambos. Em [KMSP88] temos uma amostra de três documentos de cada conjunto. Usuários eram perguntados para julgar os documentos com todas as consultas, com escala de 1 a 4. O julgamento com a consulta de termos simples eram usadas como a base da individual $P(q_i \mid \text{relevância})$ e fq_i(d)s nas várias fórmulas dos dois modelos. A Tabela 3 mostra a fórmula usada para consultas complexas. Há 50 respondentes (ou usuários) que na maioria eram estudantes universitários, com vários profissionais e supervisores pessoais.

Conjunto 1 de consultas:
Consulta 1. Comida
Consulta 2. Consumidor
Consulta 3. Publicidade
Consulta 4. Tecnologia
Consulta 5. Publicidade OR Consumidor OR Comida OR Tecnologia
Consulta 6. (Publicidade OR Consumidor) AND (Comida OR Tecnologia)
Consulta 7. Publicidade AND Consumidor AND Tecnologia
Consulta 8. Publicidade AND Consumidor AND Comida

Conjunto 2 de Consultas:
Consulta 9. Poluição
Consulta 10. Regulamento
Consulta 11. Negócio
Consulta 12. Saúde
Consulta 13. Poluição OR Regulamento OR Negócio OR Saúde
Consulta 14. Poluição AND Regulamento AND Negócio AND Saúde
Consulta 15. Poluição AND Regulamento AND Negócio
Consulta 16. Poluição AND Regulamento AND Saúde

Fig. 4. Lista de consultas usadas

Há dois conjuntos de documentos (10 documentos em cada conjunto) correspondendo às duas consultas usadas para a recuperação. Cada documento foi avaliado com quatro consultas de termos simples e com quatro consultas de termos complexos de seu conjunto. Duas simplificações experimentais foram feitas. Na primeira, um conceito atômico é representado por um simples termo indexado (palavra/frase). Na segunda, conceitos complexos são representados por combinações Booleanas (AND, OR) de termos indexados correspondendo a conceitos de componentes atômicos.

4.2.5 Análise

Correlações foram computadas depois agrupando os dados em vários caminhos em ordem para analisar os dados de múltiplas perspectivas. A seguir serão descritos cada grupo e número de observações em cada.

(1) sobre todas as consultas complexas: 2 conjuntos * 50 usuários * 10 *abstracts* * 4 consultas complexas = 4000 observações.

(2) Sobre todas as consultas complexas exceto a consulta 6: conjunto 1 * (50 usuários * 10 *abstracts* * 3 consultas complexas) + conjunto 2 * (50 usuários * 10 *abstracts* * 4 consultas complexas) = 3500 observações. A consulta 6 como a única consulta que incluiu ambos operadores 'OR' e 'AND'.

(3) Sobre todas as consultas 'OR': conjunto 1 * (50 usuários * 10 *abstracts* * 1 consulta complexa) + conjunto 2 * (50 usuários * 10 *abstracts* * 1 consulta complexa) = 1000 observações.

(4) Sobre todas as consultas 'AND': conjunto 1 * (50 usuários * 10 *abstracts* * 2 consultas complexas) + conjunto 2 * (50 usuários * 10 *abstracts* * 3 consultas complexas) = 2500 observações.

(5) Sobre cada consulta complexa separadamente: 50 usuários * 10 *abstracts* = 500 observações, isto é, 8 consultas de 500 observações cada.

(6) Sobre cada *abstract* separadamente: 50 usuários * 4 consultas complexas = 200 observações, isto é, 20 *abstracts* de 200 observações cada.

(7) Sobre os 10 *abstracts* do conjunto 1: 50 usuários * 4 consultas complexas * 10 abstracts = 2000 observações.

(8) Sobre 10 *abstracts* do conjunto 2: 50 usuários * 4 consultas complexas * 10 abstracts = 2000 observações.

Em [KMSP88] é mostrada a correlação com cada perspectiva. Estas correlações foram transformadas dentro de valores *t* que eram então testados da significância de *alpha* = 0.1, 0.05, e 0.01. O resultado desta análise é então mostrado em um tabela, encontrada na referência acima.

Tabela 3. Formula usada com consultas complexas

Tipo de Consulta	nº da consulta	Fuzzy	Probabilístico
A OR B OR C OR D	5, 13	Max(a,b,c,d)	a+b+c+d-ab-ac-ad-bc-bd-cd
(A OR B) AND (C OR D)	6	Min(Max(a,b), Max(c,d))	(a+b-ab)(c+d-cd)
A AND B AND C	7,8,15,16	Min(a,b,c)	abc
A AND B AND C AND D	14	Min(a,b,c,d)	abcd

4.2.6 Resultados

A correlação da figura mostra que o modelo Fuzzy realiza significativamente melhor do que o modelo probabilístico que é percebido através do conjunto de consultas 'AND'. Nenhuma diferença significante foi percebida no conjunto de consultas 'OR'.

Foi mostrado que o desempenho do modelo probabilístico é melhor do que o modelo Fuzzy, especialmente quando o resultado da busca consiste principalmente de documentos relevantes. Em outras palavras, talvez as diferenças entre os dois modelos são enfatizadas nas buscas de alta precisão.

A análise mostra que o modelo Fuzzy é às vezes um melhor preceptor de julgamento de relevância que comparado ao modelo probabilístico. Uma implicação é que o modelo Fuzzy pode ser mais apropriado para sistemas de recuperação de documentos que o modelo probabilístico, dando que o potencial de relevância é à base da recuperação. Pode ser notado que esta conclusão aplica ao modelo probabilístico como mostrado aqui para incluir operadores Booleanos.

Uma segunda e mais global conclusão deste estudo considera o processo de tomar decisão, usadas pelos humanos quando fazendo julgamento complexo de um individual. Aparece que o processo empregado é mais similar à

construção e manipulação da teoria dos conjuntos Fuzzy que o correspondente probabilístico. Esta conclusão é empírica apesar do fato do modelo probabilístico ter uma teoria mais poderosa que o modelo Fuzzy. Enquanto os conceitos estão sendo formados, o modelo Fuzzy é o melhor aproximador do processo humano.

Desde que os sistemas de recuperação fazem o trabalho pela combinação de termos e lógica (com ou sem pesos), esta mistura é importante para os desenvolvedores de sistemas. A implicação é que o sistema pode usar mais ANDs do que as pessoas fazem. Quando usuários usam um AND, eles não o fazem tão rigidamente. Certamente, é melhor para um documento ter ambas as palavras, mas falhando em ter ambas não é tão mau como a probabilidade da lógica OR destaca.

Para julgarmos relevância de sistema e usuário, precisa-se aprender mais sobre o processo humano.

4.3 Um sistema de recuperação de informação usando matriz de conexão de palavras-chave e um método de aprendizado

Pelo uso de matriz de conexão de palavras-chave, documentos são classificados de acordo com sua relevância a consulta do usuário. Em adição, a matriz de conexão de palavras-chave ajuda o usuário a formular a consulta apropriada ao assunto que ele ou ela deseja recuperar [OMK89].

Comparada a outros sistemas, uma nova função importante no método é a inclusão do de aprendizado. Porque o valor de pertinência é inicialmente apontado baseado em informações estatísticas, eles nem sempre fazem acordo com a medida de similaridade do usuário. Para resolver este problema, é proposto um método de aprendizado de valor de pertinência. Durante o processo

de aprendizado, os valores são mudados para reduzir as diferenças entre o valor de pertinência baseado em informação estatística, e a avaliação do usuário.

4.3.1 Matriz de conexão de palavras-chave

Uma matriz de conexão de palavras-chave é composta de um número de palavras-chave e suas afinidades, onde o valor de afinidade representa a similaridade conceitual entre duas palavras-chave.

A matriz de conexão de palavras-chave é representada pela matriz *W K x K* onde *K* é igual ao número de palavras-chave.

O valor de afinidade é restrito ao intervalo [0,1], onde "0" indica nenhuma afinidade entre duas palavras-chave e "1" indica a afinidade total possível. O valor de afinidade inicial é apontado baseado na suposição que a maioria dos documentos em que duas palavras-chave co-ocorrem, a maioria deles relatam cada outro. W_{ij}^{*} , o valor de afinidade inicial entre a *i*-ésima e a *j*-ésima palavra-chave, é dada como

$$W_{ij}^{*} = \begin{cases} \dfrac{N_{ij}}{N_1 + N_j - N_{ij}}, i \neq j, \\ 1, i = j, \end{cases}$$

onde N_{ij} é o número de documentos contendo tanto a *i*-ésima quanto a *j*-ésima palavra-chave, respectivamente. Esta fórmula determina a co-ocorrência de duas palavras-chave nos documentos.

4.3.2 Recuperação Fuzzy usando a matriz de conexão de palavras-chave

Aqui será descrito um sistema de recuperação de documentos Fuzzy usando uma matriz de conexão de palavras-chave [OMK89].

4.3.2.1. Revisão do método de recuperação Fuzzy

A consulta contém algumas palavras-chave e possivelmente operadores lógicos tais como AND (conjunção), OR (disjunção) e NOT (negação). A consulta pode ser convertida para uma forma normal conjuntiva, consistindo de sub-consultas incluindo somente OR e NOT, pela repetitividade aplicando as regras básicas da álgebra Booleana. A consulta na forma normal conjuntiva é escrita como

Consulta = sub-consulta(1) $\wedge \cdots \wedge$ sub-consulta(N),

Sub-consulta(h) = $K_1 \vee \cdots \vee K_{n_h} \vee \neg K_{n_h+1} \vee \cdots \vee \neg K_{n_h+m_h}$,

Onde $\wedge$, $\vee$ e $\neg$ representam AND, OR e NOT, K_i representa a *i*-ésima palavra-chave na consulta. Além disso, N $\geq$ 1 da consulta, $n_h \geq 0$, $m_h \geq 0$ e $n_h + m_h \geq 1$ da *h*-ésima sub-consulta. A *h*-ésima sub-consulta pode ser representada por dois conjuntos $Q(h)^+$ e $Q(h)^-$, onde $Q(h)^+$ denota o conjunto de palavras-chave sem NOT, e $Q(h)^-$ denota o conjunto de palavras-chave com NOT. Deste modo, nenhuma palavra-chave é incluída nem em $Q(h)^+$ nem em $Q(h)^-$.

O resultado da recuperação de métodos ordinários é como segue. O resultado da h-ésima sub-consulta é dado por

Sub-resultado(*h*) = $D(K_1) \cup \cdots \cup D(K_{n_h}) \cup \overline{D(K_{n_h+1})} \cup \cdots \cup \overline{D(K_{n_h+m_h})}$

Onde $D(K)$ representa o conjunto de documentos indexados pela palavra-chave K, $D_1 \cup D_2$ é a união de dois conjuntos D_1 e D_2. O operador OR na consulta requer a união dos correspondentes conjuntos de documentos, e o operador NOT requer o complemento. O resultado pode então ser representado como

$$\text{Resultado} = \text{sub-resultado}(1) \cap \cdots \cap \text{sub-resultado}(N)$$

Onde $D_1 \cap D_2$ é a interseção de dois conjuntos D_1 e D_2. O operados AND na consulta necessita a interseção dos conjuntos correspondentes. O resultado é um conjunto *crisp*[*] com todos os elementos.

Nosso sistema pretende aumentar a interface com o usuário de um sistema de recuperação *crisp* convencional. O valor de afinidade de cada documento representa suas relevâncias como um documento recuperado. O valor de afinidade do *i*-ésimo documento d_i é escrito como r_i de simplicidade

$$r_i = \mu_{\text{Resultado}}(d_i).$$

4.3.2.2 Computação do valor de afinidade usando matriz de conexão de palavras-chave

A computação do valor de relevância, em outras palavras o valor de afinidade, é computado seguindo três passos:

(a) Geração de índices Fuzzy

(b) Computação dos valores de relevância de cada sub-consulta

(c) Computação de todo o valor de relevância

Nas subseções seguintes, estes passos serão explicados em detalhes.

[*] Conjuntos tradicionais.

4.3.2.3 Geração de índices Fuzzy

Sendo que A_i denota o conjunto *crisp* de palavras-chave indexadas para o *i*-ésimo documento. Em nosso método, indexação é fazer Fuzzy pela matriz de conexão de palavras-chave como segue. R_{ij} representa a força do relacionamento entre o *i*-ésimo documento e a *j*-ésima palavra-chave é definido como

$$R_{ij} = \bigoplus_{K_k \in A_i} W_{jk}$$

Onde W_{jk} é o valor da relação entre a *j*-ésima e a *k*-ésima palavra-chave na matriz de conexão de palavras-chave. $\oplus$ denota a soma algébrica definida por $\oplus_i X_i = 1 - \prod_i (1 - X_i)$. A equação (2) torna-se

$$R_{ij} = 1 - \prod_{K_k \in A_i} (1 - W_{jk})$$

R_{ij} é referido para como valor de afinidade do *i*-ésimo documento com um índice Fuzzy para a *j*-ésima palavra-chave.

4.3.2.4 Computação do valor de relevância de cada sub-consulta

Na teoria dos conjuntos Fuzzy, a união de dois conjuntos A e B é às vezes especificado como

$$\mu_{A \cup B}(x) = \mu_A(x) \oplus \mu_B(x) = 1 - \mu_{\bar{A}}(x) \cdot \mu_{\bar{B}}(x)$$

onde $\mu_A(x)$ e $\mu_B(x)$ representa o valor de afinidade do elemento *x* nos conjuntos Fuzzy, *A* e *B*. Além de MAX, que é usualmente aplicada na união, a soma algébrica é usada para computar a necessitada derivativa do método de aprendizado como descrito na próxima seção.

O complemento do conjunto Fuzzy *A* é definido como

$$\mu_{\bar{A}}(x) = 1 - \mu_A(x).$$

O valor de relevância das sub-consultas são computadas usando

$$r_i(h) = \left(\bigoplus_{K_j \in Q(h)^+} R_{ij} \right) \oplus \left\{ \bigoplus_{K_j \in Q(h)^-} (1 - R_{ij}) \right\}$$
$$= 1 - \left(\prod_{K_j \in Q(h)^+} S_{ij} \right) \left(\prod_{K_j \in Q(h)^-} R_{ij} \right)$$

Onde

$$S_{ij} = 1 - R_{ij} = \prod_{K_k \in A_i} (1 - W_{jk})$$

4.3.2.5 Computação do valor de relevância total

Depois que todos valores das sub-consultas são determinados, o valor de relevância total pode ser computado. Na teoria de conjuntos Fuzzy, a interseção de dois conjuntos é às vezes definido como

$$\mu_{A \cap B}(x) = \mu_A(x) \cdot \mu_B(x).$$

O produto algébrico é usado para a interseção em vez de MIN exatamente como a soma algébrica era usada para a união. Deste modo, o valor da relevância do *i*-ésimo documento é computado por

$$r_i = \prod_{h=1}^{N} r_i(h).$$

4.3.3 Refinamento da consulta

O processo de recuperação é interativo e não-determinístico, e, deste modo, é muito importante ajudar o usuário a fazer melhor as consultas. Desde que a

matriz de conexão de palavra-chave representa a similaridade entre duas palavras-chave, ela pode ser usada para refinar a consulta.

No processo de criar uma consulta, o usuário pode requerer o sistema para listar palavras-chave em ordem da intensidade de sua pertinência à consulta. A intensidade da pertinência é calculada no mesmo caminho que o valor da relevância de um documento. T_i, o valor de relevância como uma palavra-chave da *i*-ésima palavra-chave, é computada como:

$$T_i(h) = 1 - \left(\prod_{K_j \in Q(h)^+} W_{ij} \right) \left\{ \prod_{K_j \in Q(h)^-} (1 - W_{ij}) \right\},$$

$$T_i = \sum_{h=1}^{N} T_i(h).$$

4.3.4 Método de aprendizado da matriz de conexão de palavras-chave

Na seção 4.4.1 é apresentado um método de aprendizado formado por várias equações. Na seção 4.4.2 é apresentada a computação da derivada parcial de uma sub-rotina e na seção 4.4.3 é apresentada a computação da derivada parcial de uma consulta, sendo que estas computações são descrições de equações. Estas três seções estão em [OMK89].

4.3.5 Avaliação do desempenho

4.3.5.1. Modelo de avaliação

A performance do sistema de recuperação de informação é normalmente medido com as duas fórmulas seguintes, a razão de revocação *R* e a razão de precisão *P*:

$$R = \frac{\#(\text{Documentos Relevantes no resultado})}{\#(\text{Total de documentos relevantes})},$$

$$P = \frac{\#(\text{Documentos Relevantes no resultado})}{\#(\text{Documentos no resultado})},$$

A razão de revocação representa como alguns dos documentos relevantes são recuperados, enquanto a razão de precisão representa quantos dos documentos recuperados são relevantes. Nós temos que converter o resultado da recuperação Fuzzy para *crisp.*

Em [OMK89] são apresentados alguns gráficos que apontam os resultados das performances. No primeiro há uma comparação com o método *crisp*, e no segundo há a efetividade do aprendizado.

O método de recuperação de documentos Fuzzy foi extendido usando uma matriz de conexão de palavras-chave. Algumas consultas compostas de palavras-chave e AND, OR e/ou NOT são processados no novo método, e o método de aprendizado é modificado pelo julgamento Fuzzy tão bem quanto de consultas complexas.

A medição da razão de revocação e razão de precisão verificou a efetividade do método proposto. A Tabela 4 resume a performance, comparado ao método *crisp* convencional. Igualmente sem o aprendizado, a razão de revocação aumenta embora a razão de precisão diminua levemente. Ambas as razões eram melhoradas depois da incorporação do método de aprendizado.

Tabela 4. Comparação do método proposto usando matriz de conexão de palavras-chave e o método crisp.

	Razão de Revocação	Razão de Precisão
Método *Crisp*	41%	43%
Método Proposto (antes do aprendizado)	56%	40%
Método Proposto (depois do aprendizado)	75%	50%

CAPÍTULO 5
PROPOSTA DE UMA MÁQUINA DE BUSCA EFICIENTE PARA DOCUMENTOS NA WEB USANDO LÓGICA FUZZY

Todo o processo de Recuperação de Informação (RI) foi descrito no capítulo 2. Neste capítulo está descrito tudo o que é necessário para se entender o funcionamento de uma máquina de busca. A seguir será proposta a máquina de busca para documentos da Web, sendo que, além dessa forma descritiva, a figura 5, também ajuda a entendê-la:

- A primeira coisa a ser feito é buscar páginas na Web para que elas possam ser recuperadas mais tarde. Na máquina de busca proposta, usa-se uma arquitetura centralizada, isto é, um software chamado *Crawler* fica responsável por esta tarefa. Os *Crawlers* atravessam a Web enviando páginas novas ou atualizadas para o principal servidor onde elas estão indexadas.
- O passo seguinte é justamente fazer a indexação dessas páginas capturadas pelos *crawlers*. Ainda no indexador, é feita a eliminação das *stopwords*, dos espaços e da pontuação.
- Com isso cria-se o arquivo invertido, que como já foi dito no capítulo 2, é uma lista de palavras, sendo que cada uma delas aponta para uma página.
- Depois de criado o arquivo invertido, o administrador do banco de dados já pode que especificar os documentos que serão usados, as operações que serão realizadas no texto, e o texto modelo.
- A interface de consulta é como nas tradicionais. Há uma caixa onde uma ou mais palavras podem ser digitadas. Essas palavras são palavras-chave e devem ser de forma que o sistema entenda o que deve ser recuperado.

- No processamento, foi escolhida a aplicação do sistema de recuperação de informação usando matriz de conexão de palavras-chave e um método de aprendizado, mostrado na seção 4.3. Nesse método, uma importante vantagem que foi mostrada, é que há uma ajuda ao usuário para a formulação da consulta. Com isso, o grande problema de formular a consulta fica minimizado. Além desse, os outros motivos que levaram à escolha dessa aplicação foram os resultados encontrados na avaliação de desempenho (*recall – precision*). Com essa avaliação fica claro que a matriz de conexão de palavras-chave ajuda a recuperar mais documentos relevantes.
- A interface de resposta usualmente consistirá de uma lista contendo em cada tela as dez páginas com maior relevância, sendo que cada página será especificada por um título, uma pequena descrição e sua URL.
- A ordenação é feita sem acesso ao texto, somente ao índice. As páginas estarão ordenadas de forma decrescente em relação às suas relevâncias. Esta ordenação normalmente usa o modelo Booleano.

A Figura 5 mostra a proposta desta máquina de busca de uma forma fácil e clara:

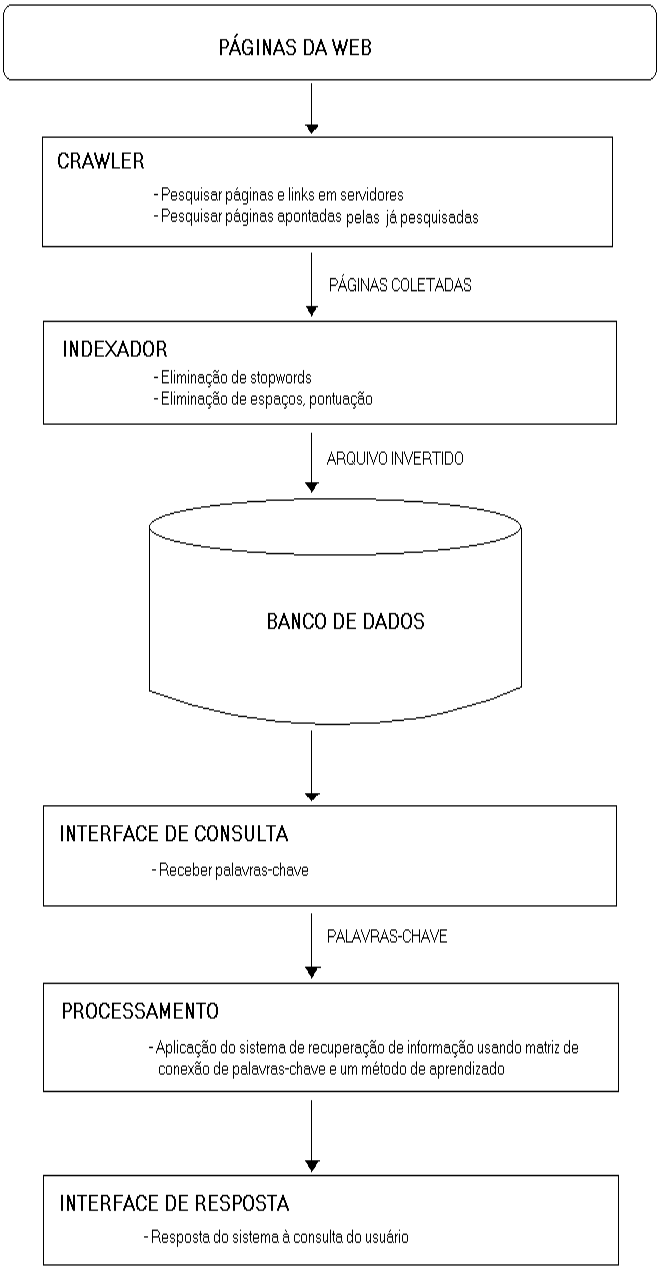

Figura 5: Proposta de uma máquina de busca para Web.

CAPÍTULO 6
CONCLUSÕES

Cada vez mais fica clara a importância dos Sistemas de Recuperação de informações nos dias de hoje, principalmente os direcionados para a Web. Várias técnicas são usadas para que estes sistemas fiquem mais rápidos e mais precisos.

Uma das técnicas que há algum tempo já está sendo estudada e que já ajuda muito a melhorar o desempenho dos Sistemas de Recuperação de Informações é a lógica Fuzzy. Esta lógica pode ser usada de várias formas dentro dos sistemas: no julgamento da relevância, na recuperação dos documentos, na formulação de consultas, etc.

O estudo feito mostrou um estudo de três aplicações da lógica Fuzzy nos Sistemas de Recuperação de Informações. Dessas três aplicações, o uso da matriz de conexão de palavras-chave e um método de aprendizado é um bom exemplo do uso da lógica Fuzzy em Recuperação de Informação, onde o resultado obtidos com testes depois do uso da matriz é bem melhor do que sem ela.

6.1 Trabalhos Futuros

Sugere-se, para trabalhos futuros, a implementação desta máquina de busca proposta neste trabalho. A implementação, utilizando a aplicação da matriz de conexão de palavras-chave, pode comprovar, ou não, tudo o que foi concluído no capítulo 4.

Sugere-se ainda o estudo de outras técnicas para que, os Sistemas de Recuperação de Informações não parem de se desenvolverem melhorando suas eficiências.

REFERÊNCIAS BIBLIOGRÁFICAS:

[ARBE94] ARBEX, ROBERTO TAIAR. **Controle Fuzzy: Circuito e aplicações** Revista Instec, junho/94, pg. 18-22 URL: http://po1.pep.ufrj.br/~mario_jo/tesim/fuzzy.htm

[BNW96] BAUER, P; NOUAK, S; WINKLER, R. **A brief course in Fuzzy Logic and Fuzzy Control**. 1996. URL: http://www.flll.uni-linz.ac.at/pdw/fuzzy/fuzzy.html

[BaRi99] BAEZA-YATES, R; RIBEIRO-NETO, B. **Modern Information Retrieval.** New York: Addison Wesley, 1999, 513p.

[BRU85] BRULE, JAMES F. **Fuzzy Systems – A Tutorial.** 1985. URL: www.austinlinks.com/Fuzzy.

[BXR02] BELCHIOR, A. D.; XEXÉO,G.B.; ROCHA, A. R. C. **Aplicação da teoria fuzzy em requisitos de qualidade de software**. Programa de Engenharia de Sistemas e Computação - COPPE/UFRJ. Acessado em Junho de 2002. URL: http://www.cos.ufrj.br/~xexeo/artigos/clei96/clei1.htm.

[CPDUM94] C. MIC BOWMAN, PETER B. DANZING, DARREN R. HARDY, UDI MANBER, e MICHAEL F. SCHWARTZ. **The Harvest Information Discovery and Access system.** In *Proc. 2nd Int. WWW Conf*., páginas 763-771, Outubro 1994.

[CROF86] CROFT, W. B. **Boolean queries and term dependencies in probabilistic retrieval models.** Journal of the American Society for Information Science., 37, 71-77.

[GSal71] G. SALTON. **The SMART Retrieval System – Experiments in Automatic Document Processing**. Prentice Hall Inc., Englewood Cliffs, NJ, 1971.

[GSME68] G. SALTON and M. E. LESK. **Computer evaluation of indexing and text processing.** *Journal of the ACM,* 15(1):8-36, January 1968.

[JWSP84] JAMES BOYLE, WILLIAM OGDEN, STEVEN UHLIR, and PATRICIA WILSON. QMF usability: How it really happened. In *Proc. Of IFIP INTERACT'84*: **Humam-Computer Interraction,** páginas 877-882, 1984.

[KMSP88] KOLL, M.; SRINIVASAN, P. **Fuzzy versus probabilistic models for user relevance judgments.** Personal Library Software Incorporated, Rockville, & University of Lowa, Lowa City, Lowa.

[KlYu95] KLIR, G. J.; YUAN, B. **Fuzzy Sets and Fuzzy Logic: theory and applications.** Upper Saddle River, New Jersey: Prentice Hall P T R, 1995, 574p.

[OMK89] OGAWA, Y.; MORITA, T.; KOBAYASHI, K. **A fuzzy document retrieval system using tha keyword connection matrix and a learning method.** Research and Development Center, RICOH Co., Ltd., 16-1 Shinei-cho, Kohoku-ku, Yokohama, Japan, 1989.

[RADE77] RADECKI, T. **Mathematical model of time-effective information retrieval system based on the theory of fuzzy sets**, Inform. Process. Management 13 (1977) 109-116.

[RADE79] RADECKI, T. **Fuzzy set theoretical approach to document retrieval,** Inform. Process. Management 15 (1979) 247-259.

[TRY02] TAKEMURA, R. Y. **Lógica Difusa.** URL: http://www.din.uem.br/ia/controle/fuz_prin.htm. Acessado em Junho de 2002.

[XEXE98] XEXÉO, G. B. **Nada Nebuloso na Lógica Difusa**. URL: http://ww.cos.ufrj.br/~xexeo. 1998.

[ZADE65] ZADEH, L. A.; **Fuzzy Sets**, Inform. And Control 8 (1965), Universidade da Califórnia, Berkeley.

[ZDCK84] ZENNER, R. B. R. C.; DE CALUWE, R. M. M.; KERRE, E. E. **A new approach to information retrieval systems using fuzzy expressions.** State University of Ghent, Belgium. 1984.